

日 本 国 特 許 庁
PATENT OFFICE
JAPANESE GOVERNMENT

JC841 U.S. PTO
09/741802
12/22/00

別紙添付の書類に記載されている事項は下記の出願書類に記載されて
いる事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed
with this Office.

出 願 年 月 日
Date of Application:

2000年 8月 2日

出 願 番 号
Application Number:

特願2000-234490

出 願 人
Applicant (s):

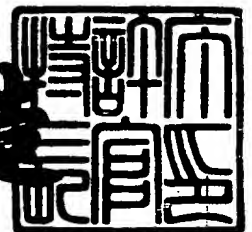
富士通株式会社

CERTIFIED COPY OF
PRIORITY DOCUMENT

2000年10月 6日

特許庁長官
Commissioner,
Patent Office

及 川 耕 造



出証番号 出証特2000-3081613

【書類名】 特許願

【整理番号】 0040582

【提出日】 平成12年 8月 2日

【あて先】 特許庁長官殿

【国際特許分類】 G06F 7/00

【発明の名称】 情報処理装置および特定用途演算命令の例外処理方法

【請求項の数】 12

【発明者】

 【住所又は居所】 神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内

 【氏名】 三宅 英雄

【発明者】

 【住所又は居所】 神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内

 【氏名】 須賀 敦浩

【発明者】

 【住所又は居所】 神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内

 【氏名】 中村 泰基

【特許出願人】

 【識別番号】 000005223

 【氏名又は名称】 富士通株式会社

【代理人】

 【識別番号】 100104190

 【弁理士】

 【氏名又は名称】 酒井 昭徳

【手数料の表示】

 【予納台帳番号】 041759

 【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 9906241

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 情報処理装置および特定用途演算命令の例外処理方法

【特許請求の範囲】

【請求項 1】 特定用途演算命令の実行中に発生した演算例外を検出し、例外処理をおこなう情報処理装置において、

プログラムの実行が中断された後、コンテキストを退避する退避手段と、
特定用途演算命令の実行中に演算例外を検出したか否かを確認する確認手段と

、
特定用途演算命令の実行中の演算例外が検出済みである場合に例外処理をおこなう例外処理手段と、

割り込みから復帰する復帰手段と、
を備えたことを特徴とする情報処理装置。

【請求項 2】 特定用途演算命令の実行中に発生する演算例外の検出をおこなう状態に設定されているか否かを確認する第 2 の確認手段をさらに備えたことを特徴とする請求項 1 に記載の情報処理装置。

【請求項 3】 ブレーク対象命令が特定用途演算命令であるか否かを確認する第 3 の確認手段をさらに備えたことを特徴とする請求項 1 に記載の情報処理装置。

【請求項 4】 特定用途演算命令の実行中に演算例外を検出したことを記憶手段に記憶し、前記記憶手段の内容を参照して演算例外が検出済みであるか未検出であるかを確認することを特徴とする請求項 1 ～ 3 のいずれか一つに記載の情報処理装置。

【請求項 5】 特定用途演算命令の実行中に発生した演算例外を検出し、例外処理をおこなう特定用途演算命令の例外処理方法において、

プログラムの実行が中断された後、コンテキストを退避する退避工程と、
特定用途演算命令の実行中に演算例外を検出したか否かを確認する確認工程と

、
特定用途演算命令の実行中の演算例外が検出済みである場合に例外処理をおこなう例外処理工程と、

割り込みから復帰する復帰工程と、

を含むことを特徴とする特定用途演算命令の例外処理方法。

【請求項 6】 特定用途演算命令の実行中に発生する演算例外の検出をおこなう状態に設定されているか否かを確認する第 2 の確認工程をさらに含むことを特徴とする請求項 5 に記載の特定用途演算命令の例外処理方法。

【請求項 7】 ブレーク対象命令が特定用途演算命令であるか否かを確認する第 3 の確認工程をさらに含むことを特徴とする請求項 5 に記載の特定用途演算命令の例外処理方法。

【請求項 8】 特定用途演算命令の実行中に演算例外を検出したことを記憶手段に記憶し、前記記憶手段の内容を参照して演算例外が検出済みであるか未検出であるかを確認することを特徴とする請求項 5 ～ 7 のいずれか一つに記載の特定用途演算命令の例外処理方法。

【請求項 9】 特定用途演算命令の実行中に発生する演算例外の検出をおこなう状態に設定されたことをレジスタまたはメモリのフラグに記憶し、前記レジスタまたは前記フラグを参照して演算例外の検出をおこなう状態に設定されているか否かを確認することを特徴とする請求項 6 に記載の特定用途演算命令の例外処理方法。

【請求項 10】 特定用途演算命令の実行中に発生する演算例外の検出をおこなう状態に設定するための命令を設け、特定用途演算命令の実行中に発生する演算例外の検出をおこなう状態に設定するための命令であるか否かを確認することを特徴とする請求項 6 に記載の特定用途演算命令の例外処理方法。

【請求項 11】 プログラムの実行を中断した命令アドレスが、特定用途演算命令の実行中に発生する演算例外の検出をおこなう対象であることを記憶手段に記憶し、前記記憶手段の内容を参照して演算例外の検出をおこなう状態に設定されているか否かを確認することを特徴とする請求項 6 に記載の特定用途演算命令の例外処理方法。

【請求項 12】 特定用途演算命令の実行中に発生する演算例外の検出をおこなう対象のブレークポイントを記憶手段に記憶し、前記記憶手段の内容を参照して演算例外の検出をおこなう状態に設定されているか否かを確認することを特

徴とする請求項6に記載の特定用途演算命令の例外処理方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、特定用途演算命令を搭載した情報処理装置および特定用途演算命令を搭載した情報処理において特定用途演算命令を実行中に演算例外が発生した場合の例外処理方法に関する。ここで、特定用途演算命令とは、特定用途に特化した演算機能を持つ命令のことである。

【0002】

【従来の技術】

特定分野のアプリケーションでは、その用途に特化した特殊な演算処理が存在することがある。その場合、演算処理を効率よくおこなうため、特定用途に特化した演算機能を持つ命令（以下、特定用途演算命令とする）を搭載した情報処理装置が使用される。このような情報処理装置において、特定用途演算命令を実行した際に、プログラムの欠陥であるバグに起因するオーバーフローやアンダーフローなどの演算例外が発生する場合がある。このようなバグをなくすため、プログラムの開発段階においてデバッグ作業がおこなわれる。

【0003】

デバッグ作業はデバッグ支援システムを用いて実施される。一般に、デバッグ支援システムには、実行中のプログラムを中断させるためのブレークポイント機能やシングルステップ機能が設けられている。デバッグの作業者は、これらの機能を利用してデバッグ対象プログラムをその実行中にいったん停止させ、その状態でレジスタやメモリの値を調べたり、あるいは変更することができる。そして、デバッグの作業者は、その停止した時点からプログラムの実行を再開させることができる。

【0004】

ここで、ブレークポイント機能の実現方法として、命令ブレーク法とソフトウェア・ブレーク法がある。命令ブレーク法では、プログラムの実行開始前に、ハードウェアに設けられたブレークポイント・レジスタに、実行を停止させたい命

令のアドレスが保持される。プログラムの実行中に、ブレークポイント・レジスタにより指定された命令が検出されると、割り込みが発生し、割り込み処理プログラムを経由してデバッグ支援プログラムに制御が渡り、デバッグの作業者にその旨が通知される。デバッグの作業者がプログラムの実行再開を指示すると、制御がデバッグ支援プログラムから元に戻される。

【0005】

図17は、命令ブレーク法による割り込み処理プログラムのフローチャートである。プログラムの実行が中断され、割り込み処理プログラムが開始されると、コンテキストを退避し（ステップS1701）、命令ブレーク処理をおこなった（ステップS1702）後、コンテキストを復元し（ステップS1703）、割り込みから復帰する（ステップS1704）。

【0006】

ソフトウェア・ブレーク法では、プログラムの実行開始前に、実行を停止させたい命令が、割り込みを発生させる命令で置き換えられる。プログラムの実行中に、その割り込みを発生させる命令が検出されると、割り込みが発生し、割り込み処理プログラムを経由してデバッグ支援プログラムに制御が渡り、デバッグの作業者にその旨が通知される。プログラムの実行再開時には、置き換えられた元の命令の実行がシミュレートされた後、制御が元に戻される。なお、割り込みを発生させる命令として、ブレークポイント命令と呼ばれる専用の命令を用意する場合もある。

【0007】

図18は、ソフトウェア・ブレーク法による割り込み処理プログラムのフローチャートである。プログラムの実行が中断され、割り込み処理プログラムが開始されると、コンテキストを退避し（ステップS1801）、ソフトウェア・ブレーク処理をおこなった（ステップS1802）後、コンテキストを復元し（ステップS1803）、割り込み復帰命令により割り込みから復帰する（ステップS1804）。

【0008】

シングルステップ機能では、デバッグ対象プログラムが停止している状態で、

デバッグの作業者がシングルステップ機能の実行を指示すると、つぎに実行すべき命令が実行された後、プログラムが停止する。シングルステップ機能の実現方法としてステップ・ブレイク法がある。ステップ・ブレイク法では、命令を実行した後に割り込みを発生させるか否かを保持するレジスタとしてシングルステップ・モード・レジスタがハードウェアに設けられる。そして、シングルステップ・モードである場合には、注目している命令が実行された後に、割り込みが発生する。割り込みが発生すると、割り込み処理プログラムを経由してデバッグ支援プログラムに制御が渡り、デバッグの作業者にその旨が通知される。プログラムの実行の再開が指示されると、制御が元に戻される。

【 0 0 0 9 】

図 1 9 は、ステップ・ブレイク法による割り込み処理プログラムのフローチャートである。プログラムの実行が中断され、割り込み処理プログラムが開始されると、コンテキストを退避し（ステップ S 1 9 0 1）、ステップ・ブレイク処理をおこなった（ステップ S 1 9 0 2）後、コンテキストを復元し（ステップ S 1 9 0 3）、割り込み復帰命令により割り込みから復帰する（ステップ S 1 9 0 4）。

【 0 0 1 0 】

【発明が解決しようとする課題】

しかしながら、従来、たとえばマルチメディア処理において頻繁に発生するデータのクリップ（ある値の範囲、たとえば 0 ～ 2 5 5 の範囲や、- 3 2 7 6 8 ～ 3 2 7 6 7 の範囲に収める）に効果的な特定用途演算命令の一つである飽和演算命令の場合、オーバーフローやアンダーフローが発生しても、演算結果が上限値や下限値でクリップされてしまうため、演算例外処理がおこなわれないという問題点があった。そのため、デバッグの作業者が、オーバーフローやアンダーフローの発生の有無を逐一確認しなければならず、デバッグ作業に多大な労力と時間がかかっていた。

【 0 0 1 1 】

本発明は、上記問題点に鑑みてなされたものであって、特定用途演算命令を有する情報処理装置であって、特定用途演算命令の実行中に発生した演算例外の例

外処理をおこなうことができる情報処理装置およびその情報処理装置の特定用途演算命令の例外処理方法を提供することを目的とする。

【0012】

【課題を解決するための手段】

上記目的を達成するため、本発明にかかる情報処理装置および特定用途演算命令の例外処理方法は、プログラムの実行が中断され、割り込み処理プログラムが開始されると、コンテキストを退避した後、特定用途演算命令の実行中に発生する演算例外の検出をおこなう状態に設定されているか否かを確認する。演算例外の検出をおこなう状態に設定されている場合には、さらに特定用途演算命令の実行中の演算例外を検出済みであるか否かを確認する。演算例外を検出済みであれば特定用途演算例外処理をおこなった後、ステップ・ブレイク処理、ソフトウェア・ブレイク処理または命令ブレイク処理をおこなう。

【0013】

そして、コンテキストを復元し、割り込みから復帰して終了する。ステップ・ブレイクの場合には、特定用途演算命令の実行中に発生する演算例外の検出をおこなう状態に設定されている場合、ブレイク対象命令が特定用途演算命令である場合にのみ、特定用途演算命令の実行中の演算例外を検出済みであるか否かを確認するようにしてもよい。

【0014】

本発明によれば、特定用途演算命令を有する情報処理装置において、ステップ・ブレイク、ソフトウェア・ブレイクまたは命令ブレイクの発生時に、特定用途演算命令によるオーバーフローなどの演算例外が発生したことを検出し、例外処理をおこなうことができる。

【0015】

【発明の実施の形態】

以下に、本発明にかかる情報処理装置および特定用途演算命令の例外処理方法の実施の形態について図面を参照しつつ詳細に説明する。

【0016】

（実施の形態1）

図 1 は、本発明の実施の形態 1 にかかる特定用途演算命令の例外処理方法を適用した情報処理装置の構成を示すブロック図である。この情報処理装置は、メモリ 2 1 0、メモリ 2 1 0 から命令語を読み出す命令読み出し部 2 2 0、命令読み出し部 2 2 0 から供給された命令を実行する命令実行部 2 3 0、および割り込み制御部 2 5 0 を備えている。また、この情報処理装置は、命令読み出し部 2 2 0、命令実行部 2 3 0 および割り込み制御部 2 5 0 からの読み書きの制御をおこなうレジスタ制御部 2 6 0 を備えている。

【 0 0 1 7 】

メモリ 2 1 0 は、外部メモリ、チップ内部に存在するメモリ回路、メモリアクセスのためのバスおよびその制御回路、キャッシュ回路などを含む。前記命令読み出し部 2 2 0 は、命令読み出し制御部 2 2 1、プログラム・カウンタ 2 2 2、命令語レジスタ 2 2 3 および命令ブレイク検出部 2 2 4 を備えている。

【 0 0 1 8 】

前記命令実行部 2 3 0 は、命令デコード部 2 3 1、ロード命令実行部 2 3 2、ストア命令実行部 2 3 3、演算命令実行部 2 3 4、その他の命令の実行部 2 3 5、浮動小数点ロード命令実行部 2 3 6、浮動小数点ストア命令実行部 2 3 7、浮動小数点演算命令実行部 2 3 8、特定用途演算命令実行部 2 3 9、特定用途演算ステータス読み書き命令実行部 2 4 0、および動作モード読み書き命令実行部 2 4 1 を備えている。

【 0 0 1 9 】

前記割り込み制御部 2 5 0 は、通常の割り込み制御部 2 5 1 を備えている。前記レジスタ制御部 2 6 0 は、COND 2 6 1、割り込みから復帰する命令アドレスを保持するレジスタ（以下、「復帰命令アドレス・レジスタ」とする）2 6 2、プロセッサの動作状態（state）を保持するレジスタ（以下、「動作レジスタ」とする）2 6 3、割り込み発生前のプロセッサの動作状態（state）を保持するレジスタ（以下、「割り込み発生前動作レジスタ」とする）2 6 4、汎用レジスタ 2 6 5、浮動小数点レジスタ 2 6 6、特定用途演算ステータス・レジスタ 2 6 7、および動作モード・レジスタ 2 6 8 を備えている。

【 0 0 2 0 】

つぎに、命令読み出し部 2 2 0 の各部について説明する。前記プログラム・カウンタ 2 2 2 は、読み出すべき命令語のアドレスを示す。前記命令読み出し制御部 2 2 1 は、プログラム・カウンタ 2 2 2 が示す命令語をメモリ 2 1 0 から読み出し、それを命令語レジスタ 2 2 3 に書き込む。また、命令実行部 2 3 0 から分岐先アドレスが供給された場合、または、割り込み制御部 2 5 0 から、発生した割り込みもしくは例外に対応する分岐先アドレスが供給された場合、命令読み出し制御部 2 2 1 はその分岐先アドレスの値をプログラム・カウンタ 2 2 2 に書き込む。それ以外の場合には、つぎの命令語を示すために、命令読み出し制御部 2 2 1 はプログラム・カウンタ 2 2 2 をインクリメントする。

【 0 0 2 1 】

また、命令読み出し制御部 2 2 1 は、命令語を読み出す際に割り込みを検出すると、割り込み制御部 2 5 0 に割り込みを検出したことを通知する。前記命令語レジスタ 2 2 3 は、命令読み出し制御部 2 2 1 により読み出された命令語を保持し、それを命令実行部 2 3 0 に供給する。

【 0 0 2 2 】

前記命令ブレーク検出部 2 2 4 は、命令語の読み出しアドレスが命令ブレークとして成立するか否かを検出する。命令ブレーク検出時には、割り込み制御部 2 5 0 に命令ブレークを検出したことを通知する。図 2 は、命令ブレーク検出部 2 2 4 の一例を示すブロック図である。命令ブレーク検出部 2 2 4 は、複数組の命令ブレークポイント・レジスタ 2 2 5 a ~ 2 2 5 d および判定器 2 2 6 a ~ 2 2 6 d と、オア回路 2 2 7 を備えている。

【 0 0 2 3 】

また、各命令ブレークポイント・レジスタ 2 2 5 a ~ 2 2 5 d には、ADDR フィールド 2 2 8 a、E フィールド 2 2 8 b および V フィールド 2 2 8 c が設けられている。ADDR フィールド 2 2 8 a は、ブレークポイント対象アドレスを保持する。E フィールド 2 2 8 b は、命令ブレーク動作が有効か否かを示し、たとえば命令ブレーク動作が無効状態のときには「0（ゼロ）」を、また有効状態のときには「1」を保持する。V フィールド 2 2 8 c は、命令ブレークを検出したか否かを示し、たとえば未検出のときには「0（ゼロ）」を、また検出済みの

ときには「1」を保持する。

【0024】

判定器226a～226dは、命令ブレイク条件が成立したか否かの判定をおこなう。各判定器226a～226dは、読み出す命令の命令アドレスと対応する命令ブレイクポイント・レジスタ225a～225dのADDRフィールド228aの値が一致した場合、対応するVフィールド228cの値を「1」に設定するとともに、オア回路227に一致信号として「1」を出力する。それによって、オア回路227から割り込み制御部250に、命令ブレイクが成立したことを通知するための割り込み通知が発行される。

【0025】

命令実行部230の各部について説明する。前記命令デコード部231は、命令読み出し部220から供給された命令を解析し、対応する命令実行部に命令を供給する。すなわち、命令デコード部231は、ロード命令をロード命令実行部232に、またストア命令をストア命令実行部233に、また演算命令および比較命令を演算命令実行部234にそれぞれ供給する。

【0026】

また、命令デコード部231は、分岐命令、条件分岐命令、割り込み復帰命令、ブレイクポイント命令、動作レジスタ263に対する読み出しまたは書き込みの命令、割り込み発生前動作レジスタ264に対する読み出しまたは書き込みの命令、ならびに命令ブレイク検出部224の命令ブレイクポイント・レジスタ225a～225dに対する読み出しまたは書き込みの命令を、その他の命令の実行部235に供給する。

【0027】

また、命令デコード部231は、浮動小数点ロード命令を浮動小数点ロード命令実行部236に、また浮動小数点ストア命令を浮動小数点ストア命令実行部237に、また浮動小数点演算命令および浮動小数点比較命令を浮動小数点演算命令実行部238に、また特定用途演算命令を特定用途演算命令実行部239にそれぞれ供給する。また、命令デコード部231は、特定用途演算ステータス・レジスタ267に対する読み出しまたは書き込みの命令を特定用途演算ステータス

読み書き命令実行部 2 4 0 に、また動作モード・レジスタ 2 6 8 に対する読み出しまたは書き込みの命令を動作モード読み書き命令実行部 2 4 1 にそれぞれ供給する。

【 0 0 2 8 】

前記特定用途演算ステータス読み書き命令実行部 2 4 0 は、命令デコード部 2 3 1 から特定用途演算ステータス・レジスタ 2 6 7 に対する読み出し命令を受け取ると、特定用途演算ステータス・レジスタ 2 6 7 を読み出し、その結果を汎用レジスタ 2 6 5 に書き込む。また、特定用途演算ステータス読み書き命令実行部 2 4 0 は、命令デコード部 2 3 1 から特定用途演算ステータス・レジスタ 2 6 7 に対する書き込み命令を受け取ると、汎用レジスタ 2 6 5 を読み出し、その結果を特定用途演算ステータス・レジスタ 2 6 7 に書き込む。

【 0 0 2 9 】

また、特定用途演算ステータス読み書き命令実行部 2 4 0 は、命令を実行する際に割り込みを検出すると、割り込み制御部 2 5 0 に割り込みを検出したことを通知する。また、特定用途演算ステータス読み書き命令実行部 2 4 0 は、命令を実行した後、シングルステップ・モードであった場合には、割り込み制御部 2 5 0 にステップ・ブレークを検出したことを通知する。

【 0 0 3 0 】

前記動作モード読み書き命令実行部 2 4 1 は、命令デコード部 2 3 1 から動作モード・レジスタ 2 6 8 に対する読み出し命令を受け取ると、動作モード・レジスタ 2 6 8 を読み出し、その結果を汎用レジスタ 2 6 5 へ書き込む。また、動作モード読み書き命令実行部 2 4 1 は、命令デコード部 2 3 1 から動作モード・レジスタ 2 6 8 に対する書き込み命令を受け取ると、汎用レジスタ 2 6 5 を読み出し、その結果を動作モード・レジスタ 2 6 8 に書き込む。

【 0 0 3 1 】

また、動作モード読み書き命令実行部 2 4 1 は、命令を実行する際に割り込みを検出すると、割り込み制御部 2 5 0 に割り込みを検出したことを通知する。また、動作モード読み書き命令実行部 2 4 1 は、命令を実行した後、シングルステップ・モードであった場合には、割り込み制御部 2 5 0 にステップ・ブレークを

検出したことを通知する。

【0032】

前記ロード命令実行部232は、命令デコード部231からロード命令を受け取ると、汎用レジスタ265から読み出した値に基づいて実効アドレスを求め、メモリ210の、その実効アドレスに対応する領域を読み出し、その結果を汎用レジスタ265に書き込む。また、ロード命令実行部232は、命令を実行する際に割り込みを検出すると、割り込み制御部250に割り込みを検出したことを通知する。また、ロード命令実行部232は、命令を実行した後、シングルステップ・モードであった場合には、割り込み制御部250にステップ・ブレークを検出したことを通知する。

【0033】

前記ストア命令実行部233は、命令デコード部231からストア命令を受け取ると、汎用レジスタ265から読み出した値に基づいて実効アドレスを求めるとともに、汎用レジスタ265を読み出し、その結果をメモリ210の、その実効アドレスに対応する領域に書き込む。また、ストア命令実行部233は、命令を実行する際に割り込みを検出すると、割り込み制御部250に割り込みを検出したことを通知する。また、ストア命令実行部233は、命令を実行した後、シングルステップ・モードであった場合には、割り込み制御部250にステップ・ブレークを検出したことを通知する。

【0034】

前記演算命令実行部234は、命令デコード部231から演算命令を受け取ると、汎用レジスタ265から読み出した値に基づいて演算をおこない、その結果を汎用レジスタ265に書き込む。また、演算命令実行部234は、命令デコード部231から比較命令を受け取ると、汎用レジスタ265から読み出した値に基づいて比較をおこない、その結果として真または偽を示す値をCOND261に書き込む。また、演算命令実行部234は、命令を実行する際に割り込みを検出すると、割り込み制御部250に割り込みを検出したことを通知する。また、演算命令実行部234は、命令を実行した後、シングルステップ・モードであった場合には、割り込み制御部250にステップ・ブレークを検出したことを通知

する。

【 0 0 3 5 】

前記その他の命令の実行部 2 3 5 は、命令デコード部 2 3 1 から分岐命令を受け取ると、分岐先アドレスを命令読み出し部 2 2 0 に供給する。また、その他の命令の実行部 2 3 5 は、命令デコード部 2 3 1 から条件分岐命令を受け取ると、COND 2 6 1 の値が真を示す値であれば、分岐先アドレスを命令読み出し部 2 2 0 に供給する。

【 0 0 3 6 】

また、その他の命令の実行部 2 3 5 は、命令デコード部 2 3 1 から割り込み復帰命令を受け取ると、割り込み発生前動作レジスタ 2 6 4 の値を動作レジスタ 2 6 3 へ書き込むとともに、復帰命令アドレス・レジスタ 2 6 2 を読み出し、その結果を分岐先アドレスとして命令読み出し部 2 2 0 に供給する。また、その他の命令の実行部 2 3 5 は、命令デコード部 2 3 1 からブレークポイント命令を受け取ると、割り込み制御部 2 5 0 にソフトウェア・ブレークが成立したことを通知する。

【 0 0 3 7 】

また、その他の命令の実行部 2 3 5 は、命令デコード部 2 3 1 から動作レジスタ 2 6 3、割り込み発生前動作レジスタ 2 6 4 または命令ブレークポイント・レジスタ 2 2 5 a ~ 2 2 5 d に対する読み出し命令を受け取ると、それぞれ動作レジスタ 2 6 3、割り込み発生前動作レジスタ 2 6 4 または命令ブレーク検出部 2 2 4 の命令ブレークポイント・レジスタ 2 2 5 a ~ 2 2 5 d を読み出し、その結果を汎用レジスタ 2 6 5 に書き込む。

【 0 0 3 8 】

また、その他の命令の実行部 2 3 5 は、命令デコード部 2 3 1 から動作レジスタ 2 6 3、割り込み発生前動作レジスタ 2 6 4 または命令ブレークポイント・レジスタ 2 2 5 a ~ 2 2 5 d に対する書き込み命令を受け取ると、汎用レジスタ 2 6 5 を読み出し、その結果をそれぞれ動作レジスタ 2 6 3、割り込み発生前動作レジスタ 2 6 4 または命令ブレーク検出部 2 2 4 の命令ブレークポイント・レジスタ 2 2 5 a ~ 2 2 5 d に書き込む。

【 0 0 3 9 】

また、その他の命令の実行部 2 3 5 は、命令を実行する際に割り込みを検出すると、割り込み制御部 2 5 0 に割り込みを検出したことを通知する。また、その他の命令の実行部 2 3 5 は、命令を実行した後、シングルステップ・モードであった場合には、割り込み制御部 2 5 0 にステップ・ブレークを検出したことを通知する。

【 0 0 4 0 】

前記浮動小数点ロード命令実行部 2 3 6 は、命令デコード部 2 3 1 から浮動小数点ロード命令を受け取ると、汎用レジスタ 2 6 5 から読み出した値に基づいて実効アドレスを求め、メモリ 2 1 0 の、その実効アドレスに対応する領域を読み出し、その結果を浮動小数点レジスタ 2 6 6 に書き込む。また、浮動小数点ロード命令実行部 2 3 6 は、命令を実行する際に割り込みを検出すると、割り込み制御部 2 5 0 に割り込みを検出したことを通知する。また、浮動小数点ロード命令実行部 2 3 6 は、命令を実行した後、シングルステップ・モードであった場合には、割り込み制御部 2 5 0 にステップ・ブレークを検出したことを通知する。

【 0 0 4 1 】

前記浮動小数点ストア命令実行部 2 3 7 は、命令デコード部 2 3 1 から浮動小数点ストア命令を受け取ると、汎用レジスタ 2 6 5 から読み出した値に基づいて実効アドレスを求めるとともに、浮動小数点レジスタ 2 6 6 を読み出し、その結果をメモリ 2 1 0 の、その実効アドレスに対応する領域に書き込む。また、浮動小数点ストア命令実行部 2 3 7 は、命令を実行する際に割り込みを検出すると、割り込み制御部 2 5 0 に割り込みを検出したことを通知する。また、浮動小数点ストア命令実行部 2 3 7 は、命令を実行した後、シングルステップ・モードであった場合には、割り込み制御部 2 5 0 にステップ・ブレークを検出したことを通知する。

【 0 0 4 2 】

前記浮動小数点演算命令実行部 2 3 8 は、命令デコード部 2 3 1 から浮動小数点演算命令を受け取ると、浮動小数点レジスタ 2 6 6 から読み出した値に基づいて演算をおこない、その結果を浮動小数点レジスタ 2 6 6 に書き込む。また、浮

動小数点演算命令実行部 2 3 8 は、命令デコード部 2 3 1 から浮動小数点比較命令を受け取ると、浮動小数点レジスタ 2 6 6 から読み出した値に基づいて比較をおこない、その結果として真または偽を示す値を COND 2 6 1 に書き込む。また、浮動小数点演算命令実行部 2 3 8 は、命令を実行する際に割り込みを検出すると、割り込み制御部 2 5 0 に割り込みを検出したことを通知する。また、浮動小数点演算命令実行部 2 3 8 は、命令を実行した後、シングルステップ・モードであった場合には、割り込み制御部 2 5 0 にステップ・ブ레이크を検出したことを通知する。

【 0 0 4 3 】

前記特定用途演算命令実行部 2 3 9 は、命令デコード部 2 3 1 から特定用途演算命令を受け取ると、浮動小数点レジスタ 2 6 6 から読み出した値に基づいて演算をおこない、その結果を浮動小数点レジスタ 2 6 6 に書き込む。また、特定用途演算命令実行部 2 3 9 は、演算中にオーバーフローなどの演算例外が発生すると、特定用途演算ステータス・レジスタ 2 6 7 に演算例外の発生を示す値を書き込む。また、特定用途演算命令実行部 2 3 9 は、命令を実行した後、シングルステップ・モードであった場合には、割り込み制御部 2 5 0 にステップ・ブ레이크を検出したことを通知する。

【 0 0 4 4 】

つぎに、レジスタ制御部 2 6 0 の各部について説明する。前記特定用途演算ステータス・レジスタ 2 6 7 は、特定用途演算命令の実行中にオーバーフローなどの演算例外が検出されたか否かを示す値を保持する。前記動作モード・レジスタ 2 6 8 は、特定用途演算命令の実行中に発生する演算例外の検出をおこなうか否かを示す値を保持する。たとえば、このレジスタ 2 6 8 の値は、特定用途演算命令の実行中の演算例外を検出しない場合には「0（ゼロ）」に設定され、一方、特定用途演算命令の実行中の演算例外を検出する場合には「1」に設定される。前記復帰命令アドレス・レジスタ 2 6 2 および前記割り込み発生前動作レジスタ 2 6 4 は、割り込み発生時に設定される。

【 0 0 4 5 】

図 3 は、動作レジスタ 2 6 3 の一例を示す構成図である。動作レジスタ 2 6 3

には、SEフィールド269aおよびSフィールド269bが設けられている。SEフィールド269aは、シングルステップ・モードであるか否かを示し、たとえば非シングルステップ・モードのときには「0（ゼロ）」を、またシングルステップ・モードのときには「1」を保持する。したがって、このSEフィールド269aの値が「1」のときには、命令を実行した後に、ステップ・ブレーク割り込みが発生する。Sフィールド269bは、スーパーバイザ状態であるか否かを示し、たとえばユーザ状態のときには「0（ゼロ）」を、またスーパーバイザ状態のときには「1」を保持する。

【0046】

図4は、割り込み発生前動作レジスタ264の一例を示す構成図である。割り込み発生前動作レジスタ264には、SEフィールド269cおよびSフィールド269dが設けられている。SEフィールド269cは、割り込み発生前の状態がシングルステップ・モードであったか否かを示し、たとえば割り込み発生前が非シングルステップ・モードのときには「0（ゼロ）」を、また割り込み発生前がシングルステップ・モードのときには「1」を保持する。Sフィールド269dは、割り込み発生前がスーパーバイザ状態であったか否かを示し、たとえば割り込み発生前がユーザ状態のときには「0（ゼロ）」を、また割り込み発生前がスーパーバイザ状態のときには「1」を保持する。

【0047】

つぎに、前記割り込み制御部250の通常の割り込み制御部251について説明する。通常の割り込み制御部251は、命令読み出し部220または命令実行部230から割り込み通知を受け取ると、復帰命令アドレス・レジスタ262に、割り込みから復帰する命令アドレスを、また割り込み発生前動作レジスタ264に割り込み発生前のプロセッサの動作状態（state）を、動作レジスタ263に、発生した割り込みに対応するプロセッサの動作状態（state）をそれぞれ書き込む。また、通常の割り込み制御部251は、命令読み出し部220に、発生した割り込みに対応する分岐先アドレスを供給する。

【0048】

つぎに、本発明の実施の形態1にかかる特定用途演算命令の例外処理方法につ

いて説明する。図 5 は、本発明の実施の形態 1 にかかる特定用途演算命令の例外処理方法のうち、ステップ・ブレイク法による割り込み処理プログラムのフローチャートである。図 5 のフローチャートにおいて、プログラムの実行が中断され、割り込み処理プログラムが開始されると、まずコンテキストを退避する（ステップ S 5 0 1）。そして、動作モード・レジスタ 2 6 8 に対する読み出し命令により動作モード・レジスタ 2 6 8 の値を読み出し、特定用途演算命令の実行中に発生する演算例外の検出をおこなう状態に設定されているか否かを確認する（ステップ S 5 0 2）。

【 0 0 4 9 】

その結果、特定用途演算命令の実行中に発生する演算例外の検出をおこなわない状態に設定されている場合（ステップ S 5 0 3：特定用途演算例外未検出動作状態）は、ステップ・ブレイク処理をおこなった（ステップ S 5 0 4）後、コンテキストを復元し（ステップ S 5 0 5）、割り込み復帰命令により割り込みから復帰し（ステップ S 5 0 6）、割り込み処理を終了する。したがって、演算例外の検出をおこなう状態に設定されていない場合には、迅速にブレイク処理をおこなって元に戻ることができる。

【 0 0 5 0 】

一方、動作モード・レジスタ 2 6 8 を確認した結果、特定用途演算命令の実行中に発生する演算例外の検出をおこなう状態に設定されている場合（ステップ S 5 0 3：特定用途演算例外検出動作状態）は、メモリ 2 1 0 からブレイク対象命令を読み出し、その命令のコードを解析してその命令が特定用途演算命令であるか否かを確認する（ステップ S 5 0 7）。その結果、ブレイク対象命令が特定用途演算命令でない場合（ステップ S 5 0 8：非特定用途演算命令）は、ステップ S 5 0 4 へ移行し、ステップ S 5 0 4～S 5 0 6 の処理をおこない、割り込み処理を終了する。したがって、特定用途演算命令でない場合には、迅速にブレイク処理をおこなって元に戻ることができる。

【 0 0 5 1 】

これに対し、ブレイク対象命令が特定用途演算命令である場合（ステップ S 5 0 8：特定用途演算命令）は、特定用途演算ステータス・レジスタ 2 6 7 に対す

る読み出し命令により特定用途演算ステータス・レジスタ 2 6 7 の値を読み出し、特定用途演算命令の実行中の演算例外を検出済みであるか否かを確認する（ステップ S 5 0 9）。その結果、演算例外を未検出であれば（ステップ S 5 1 0：演算例外未検出）、ステップ S 5 0 4 へ移行し、ステップ S 5 0 4～S 5 0 6 の処理をおこない、割り込み処理を終了する。一方、演算例外を検出済みであれば（ステップ S 5 1 0：演算例外検出済）、特定用途演算例外処理をおこなった（ステップ S 5 1 1）後、ステップ S 5 0 4 へ移行し、ステップ S 5 0 4～S 5 0 6 の処理をおこない、割り込み処理を終了する。

【0 0 5 2】

図 6 は、本発明の実施の形態 1 にかかる特定用途演算命令の例外処理方法のうち、ソフトウェア・ブレイク法による割り込み処理プログラムのフローチャートである。図 6 のフローチャートにおいて、プログラムの実行が中断され、割り込み処理プログラムが開始されると、まずコンテキストを退避する（ステップ S 6 0 1）。そして、動作モード・レジスタ 2 6 8 に対する読み出し命令により動作モード・レジスタ 2 6 8 の値を読み出し、特定用途演算命令の実行中に発生する演算例外の検出をおこなう状態に設定されているか否かを確認する（ステップ S 6 0 2）。

【0 0 5 3】

その結果、特定用途演算命令の実行中に発生する演算例外の検出をおこなわない状態に設定されている場合（ステップ S 6 0 3：特定用途演算例外未検出動作状態）は、ソフトウェア・ブレイク処理をおこなった（ステップ S 6 0 4）後、コンテキストを復元し（ステップ S 6 0 5）、割り込み復帰命令により割り込みから復帰し（ステップ S 6 0 6）、割り込み処理を終了する。

【0 0 5 4】

一方、動作モード・レジスタ 2 6 8 を確認した結果、特定用途演算命令の実行中に発生する演算例外の検出をおこなう状態に設定されている場合（ステップ S 6 0 3：特定用途演算例外検出動作状態）は、特定用途演算ステータス・レジスタ 2 6 7 に対する読み出し命令により特定用途演算ステータス・レジスタ 2 6 7 の値を読み出し、特定用途演算命令の実行中の演算例外を検出済みであるか否か

を確認する（ステップ S 6 0 7）。その結果、演算例外を未検出であれば（ステップ S 6 0 8：演算例外未検出）、ステップ S 6 0 4 へ移行し、ステップ S 6 0 4 ～ S 6 0 6 の処理をおこない、割り込み処理を終了する。これに対し、演算例外を検出済みであれば（ステップ S 6 0 8：演算例外検出済）、特定用途演算例外処理をおこなった（ステップ S 6 0 9）後、ステップ S 6 0 4 へ移行し、ステップ S 6 0 4 ～ S 6 0 6 の処理をおこない、割り込み処理を終了する。

【 0 0 5 5 】

図 7 は、本発明の実施の形態 1 にかかる特定用途演算命令の例外処理方法のうち、命令ブレイク法による割り込み処理プログラムのフローチャートである。図 7 のフローチャートにおいて、プログラムの実行が中断され、割り込み処理プログラムが開始されると、まずコンテキストを退避する（ステップ S 7 0 1）。そして、動作モード・レジスタ 2 6 8 に対する読み出し命令により動作モード・レジスタ 2 6 8 の値を読み出し、特定用途演算命令の実行中に発生する演算例外の検出をおこなう状態に設定されているか否かを確認する（ステップ S 7 0 2）。

【 0 0 5 6 】

その結果、特定用途演算命令の実行中に発生する演算例外の検出をおこなわない状態に設定されている場合（ステップ S 7 0 3：特定用途演算例外未検出動作状態）は、命令ブレイク処理をおこなった（ステップ S 7 0 4）後、コンテキストを復元し（ステップ S 7 0 5）、割り込み復帰命令により割り込みから復帰し（ステップ S 7 0 6）、割り込み処理を終了する。

【 0 0 5 7 】

一方、動作モード・レジスタ 2 6 8 を確認した結果、特定用途演算命令の実行中に発生する演算例外の検出をおこなう状態に設定されている場合（ステップ S 7 0 3：特定用途演算例外検出動作状態）は、特定用途演算ステータス・レジスタ 2 6 7 に対する読み出し命令により特定用途演算ステータス・レジスタ 2 6 7 の値を読み出し、特定用途演算命令の実行中の演算例外を検出済みであるか否かを確認する（ステップ S 7 0 7）。その結果、演算例外を未検出であれば（ステップ S 7 0 8 演算例外未検出）、ステップ S 7 0 4 へ移行し、ステップ S 7 0 4 ～ S 7 0 6 の処理をおこない、割り込み処理を終了する。これに対し、演算例外

を検出済みであれば（ステップ S 7 0 8：演算例外検出済）、特定用途演算例外処理をおこなった（ステップ S 7 0 9）後、ステップ S 7 0 4 へ移行し、ステップ S 7 0 4 ～ S 7 0 6 の処理をおこない、割り込み処理を終了する。

【 0 0 5 8 】

上述した実施の形態 1 によれば、特定用途演算命令を有する情報処理装置において、ステップ・ブレーク、ソフトウェア・ブレーク、命令ブレークのいずれが発生した場合でも、特定用途演算命令によるオーバーフローなどの演算例外が発生したことを検出し、例外処理をおこなうことができる。したがって、デバッグ作業の労力が軽減されるとともに、デバッグ作業に要する時間が短縮されるので、プログラムの開発期間が短縮されるという効果が得られる。

【 0 0 5 9 】

なお、上述した実施の形態 1 においては、動作モード・レジスタ 2 6 8 を設け、このレジスタ 2 6 8 に、特定用途演算命令の実行中に発生する演算例外を検出するか否かの情報を保持させているが、これに限らず、メモリ 2 1 0 にフラグを設け、そのフラグに同様の情報を保持させるようにしてもよい。

【 0 0 6 0 】

（実施の形態 2）

図 8 は、本発明の実施の形態 2 にかかる特定用途演算命令の例外処理方法を適用した情報処理装置の構成を示すブロック図である。この情報処理装置が実施の形態 1 と異なるのはつぎの 4 点である。すなわち、第 1 に、特定用途演算命令の実行中に発生する演算例外の検出をおこなう状態に設定するための命令として特定用途演算例外検出ブレークポイント命令を設けたことである。第 2 に、実施の形態 1 の命令実行部 2 3 0 の代わりに設けられた命令実行部 3 3 0 が特定用途演算例外検出ブレークポイント命令の実行部 3 3 1 を有していることである。第 3 に、その命令実行部 3 3 0 には動作モード読み書き命令実行部 2 4 1 が設けられていないことである。第 4 に、実施の形態 1 のレジスタ制御部 2 6 0 の代わりに設けられたレジスタ制御部 3 6 0 に、動作モード・レジスタ 2 6 8 が設けられていないことである。

【 0 0 6 1 】

その他の構成については、実施の形態 1 と同様であるため、実施の形態 1 と同様の構成については、実施の形態 1 と同じ符号を付して説明を省略する。なお、実施の形態 2 では、上述した実施の形態 1 の説明中の動作モード読み書き命令実行部 2 4 1 および動作モード・レジスタ 2 6 8 に関する説明は適用されないものとする。

【0 0 6 2】

前記命令実行部 3 3 0 は、命令デコード部 2 3 1、ロード命令実行部 2 3 2、ストア命令実行部 2 3 3、演算命令実行部 2 3 4、その他の命令の実行部 2 3 5、浮動小数点ロード命令実行部 2 3 6、浮動小数点ストア命令実行部 2 3 7、浮動小数点演算命令実行部 2 3 8、特定用途演算命令実行部 2 3 9、特定用途演算ステータス読み書き命令実行部 2 4 0、および特定用途演算例外検出ブレークポイント命令の実行部 3 3 1 を備えている。

【0 0 6 3】

前記命令デコード部 2 3 1 は、実施の形態 1 で説明した機能のほかに、命令読み出し部 2 2 0 から供給された命令を解析した結果、その命令が特定用途演算例外検出ブレークポイント命令である場合には、その命令を特定用途演算例外検出ブレークポイント命令の実行部 3 3 1 に供給する機能を有している。

【0 0 6 4】

前記特定用途演算例外検出ブレークポイント命令の実行部 3 3 1 は、命令デコード部 2 3 1 から特定用途演算例外検出ブレークポイント命令を受け取ると、割り込み制御部 2 5 0 にソフトウェア・ブレークが成立したことを通知する。また、特定用途演算例外検出ブレークポイント命令の実行部 3 3 1 は、命令を実行する際に割り込みを検出すると、割り込み制御部 2 5 0 に割り込みを検出したことを通知する。また、特定用途演算例外検出ブレークポイント命令の実行部 3 3 1 は、命令を実行した後、シングルステップ・モードであった場合には、割り込み制御部 2 5 0 にステップ・ブレークを検出したことを通知する。

【0 0 6 5】

前記レジスタ制御部 3 6 0 は、COND 2 6 1、復帰命令アドレス・レジスタ 2 6 2、動作レジスタ 2 6 3、割り込み発生前動作レジスタ 2 6 4、汎用レジス

タ 2 6 5、浮動小数点レジスタ 2 6 6、および特定用途演算ステータス・レジスタ 2 6 7を備えている。

【 0 0 6 6 】

つぎに、本発明の実施の形態 2 にかかる特定用途演算命令の例外処理方法について説明する。図 9 は、本発明の実施の形態 2 にかかる特定用途演算命令の例外処理方法のうち、ソフトウェア・ブレーク法による割り込み処理プログラムのフローチャートである。図 9 のフローチャートにおいて、プログラムの実行が中断され、割り込み処理プログラムが開始されると、まずコンテキストを退避する（ステップ S 9 0 1）。そして、メモリ 2 1 0 からソフトウェア・ブレークを発生させた命令を読み出し、その命令のコードを解析してその命令が特定用途演算例外検出ブレークポイント命令であるか否かを確認する（ステップ S 9 0 2）。

【 0 0 6 7 】

その結果、特定用途演算例外検出ブレークポイント命令でない場合、すなわち単なるブレークポイント命令である場合（ステップ S 9 0 3：ブレークポイント命令）は、ソフトウェア・ブレーク処理をおこなった（ステップ S 9 0 4）後、コンテキストを復元し（ステップ S 9 0 5）、割り込み復帰命令により割り込みから復帰し（ステップ S 9 0 6）、割り込み処理を終了する。

【 0 0 6 8 】

一方、ソフトウェア・ブレークを発生させた命令が特定用途演算例外検出ブレークポイント命令である場合（ステップ S 9 0 3：特定用途演算例外検出ブレークポイント命令）は、特定用途演算ステータス・レジスタ 2 6 7 に対する読み出し命令により特定用途演算ステータス・レジスタ 2 6 7 の値を読み出し、特定用途演算命令の実行中の演算例外を検出済みであるか否かを確認する（ステップ S 9 0 7）。その結果、演算例外を未検出であれば（ステップ S 9 0 8：演算例外未検出）、ステップ S 9 0 4 へ移行し、ステップ S 9 0 4 ～ S 9 0 6 の処理をおこない、割り込み処理を終了する。これに対し、演算例外を検出済みであれば（ステップ S 9 0 8：演算例外検出済）、特定用途演算例外処理をおこなった（ステップ S 9 0 9）後、ステップ S 9 0 4 へ移行し、ステップ S 9 0 4 ～ S 9 0 6 の処理をおこない、割り込み処理を終了する。

【 0 0 6 9 】

実施の形態 2 におけるステップ・ブレイク法および命令ブレイク法による割り込み処理プログラムは、それぞれ図 1 9 および図 1 7 に示す従来のフローチャートと同様であるため、図示および説明を省略する。

【 0 0 7 0 】

上述した実施の形態 2 によれば、特定用途演算命令を有する情報処理装置において、ソフトウェア・ブレイクが発生した場合、特定用途演算命令によるオーバーフローなどの演算例外が発生したことを検出し、例外処理をおこなうことができる。したがって、デバッグ作業の労力が軽減されるとともに、デバッグ作業に要する時間が短縮されるので、プログラムの開発期間が短縮されるという効果が得られる。

【 0 0 7 1 】

なお、上述した実施の形態 2 においては、特定用途演算例外検出ブレイクポイント命令を設け、ソフトウェア・ブレイクが発生させた命令が特定用途演算例外検出ブレイクポイント命令であるかまたは単なるブレイクポイント命令であるかによって、特定用途演算命令の実行中に発生する演算例外の検出をおこなうか否かを設定するようにしているが、これには限らない。たとえば、ブレイクポイント命令の命令語に含まれるフィールドが示す値に基づいて、特定用途演算命令の実行中に発生する演算例外の検出をおこなうか否かを設定するようにしてもよい。

【 0 0 7 2 】

(実施の形態 3)

図 1 0 は、本発明の実施の形態 3 にかかる特定用途演算命令の例外処理方法を適用した情報処理装置の構成を示すブロック図である。この情報処理装置が実施の形態 1 と異なるのはつぎの 3 点である。すなわち、第 1 に、実施の形態 1 の命令読み出し部 2 2 0 の代わりに設けられた命令読み出し部 4 2 0 が、命令ブレイクポイント・レジスタの構成が異なる命令ブレイク検出部 4 2 4 を有していることである。第 2 に、実施の形態 1 の命令実行部 2 3 0 の代わりに設けられた命令実行部 4 3 0 には動作モード読み書き命令実行部 2 4 1 が設けられていないこと

である。第 3 に、実施の形態 1 のレジスタ制御部 2 6 0 の代わりに実施の形態 2 と同じ構成のレジスタ制御部 3 6 0 が設けられていることである。

【0 0 7 3】

その他の構成については、実施の形態 1 と同様であるため、実施の形態 1 と同様の構成については、実施の形態 1 と同じ符号を付して説明を省略する。また、レジスタ制御部 3 6 0 についても、実施の形態 2 の説明と重複するため省略する、実施の形態 3 では、上述した実施の形態 1 の説明中の動作モード読み書き命令実行部 2 4 1 および動作モード・レジスタ 2 6 8 に関する説明は適用されないものとする。

【0 0 7 4】

前記命令読み出し部 4 2 0 は、命令読み出し制御部 2 2 1、プログラム・カウンタ 2 2 2、命令語レジスタ 2 2 3 および命令ブレーク検出部 4 2 4 を備えている。図 1 1 は、命令ブレーク検出部 4 2 4 の一例を示すブロック図である。この命令ブレーク検出部 4 2 4 が実施の形態 1 と異なるのは、各命令ブレークポイント・レジスタ 4 2 5 a ～ 4 2 5 d には、ADDR フィールド 2 2 8 a、E フィールド 2 2 8 b および V フィールド 2 2 8 c のほかに M フィールド 4 2 8 が設けられていることである。命令ブレーク検出部 4 2 4 のその他の構成は実施の形態 1 の命令ブレーク検出部 2 2 4 と同じである。M フィールド 4 2 8 は、対応する命令ブレークが、特定用途演算命令の実行中に発生する演算例外の検出をおこなう対象であるか否かを示し、対象でない場合には「0（ゼロ）」を、また対象である場合には「1」を保持する。

【0 0 7 5】

前記命令実行部 4 3 0 は、命令デコード部 2 3 1、ロード命令実行部 2 3 2、ストア命令実行部 2 3 3、演算命令実行部 2 3 4、その他の命令の実行部 2 3 5、浮動小数点ロード命令実行部 2 3 6、浮動小数点ストア命令実行部 2 3 7、浮動小数点演算命令実行部 2 3 8、特定用途演算命令実行部 2 3 9、および特定用途演算ステータス読み書き命令実行部 2 4 0 を備えている。

【0 0 7 6】

つぎに、本発明の実施の形態 3 にかかる特定用途演算命令の例外処理方法につ

いて説明する。図 1 2 は、本発明の実施の形態 3 にかかる特定用途演算命令の例外処理方法のうち、命令ブレイク法による割り込み処理プログラムのフローチャートである。図 1 2 のフローチャートにおいて、プログラムの実行が中断され、割り込み処理プログラムが開始されると、まずコンテキストを退避する（ステップ S 1 2 0 1）。そして、命令ブレイクポイント・レジスタ 4 2 5 a ~ 4 2 5 d に対する読み出し命令により対応する命令ブレイクポイント・レジスタ 4 2 5 a ~ 4 2 5 d の M フィールド 4 2 8 の値を読み出し、命令ブレイクが、特定用途演算命令の実行中に発生する演算例外の検出をおこなう対象であるか否かを確認する（ステップ S 1 2 0 2）。その結果、命令ブレイクが演算例外の検出対象でない場合（ステップ S 1 2 0 3：特定用途演算例外検出動作対象外）は、命令ブレイク処理をおこなった（ステップ S 1 2 0 4）後、コンテキストを復元し（ステップ S 1 2 0 5）、割り込み復帰命令により割り込みから復帰し（ステップ S 1 2 0 6）、割り込み処理を終了する。

【 0 0 7 7 】

一方、対応する命令ブレイクが演算例外の検出対象である場合（ステップ S 1 2 0 3：特定用途演算例外検出動作対象）は、特定用途演算ステータス・レジスタ 2 6 7 に対する読み出し命令により特定用途演算ステータス・レジスタ 2 6 7 の値を読み出し、特定用途演算命令の実行中の演算例外を検出済みであるか否かを確認する（ステップ S 1 2 0 7）。その結果、演算例外を未検出であれば（ステップ S 1 2 0 8：演算例外未検出）、ステップ S 1 2 0 4 へ移行し、ステップ S 1 2 0 4 ~ S 1 2 0 6 の処理をおこない、割り込み処理を終了する。これに対し、演算例外を検出済みであれば（ステップ S 1 2 0 8：演算例外検出済）、特定用途演算例外処理をおこなった（ステップ S 1 2 0 9）後、ステップ S 1 2 0 4 へ移行し、ステップ S 1 2 0 4 ~ S 1 2 0 6 の処理をおこない、割り込み処理を終了する。

【 0 0 7 8 】

実施の形態 3 におけるステップ・ブレイク法およびソフトウェア・ブレイク法による割り込み処理プログラムは、それぞれ図 1 9 および図 1 8 に示す従来のフローチャートと同様であるため、図示および説明を省略する。

【 0 0 7 9 】

上述した実施の形態 3 によれば、特定用途演算命令を有する情報処理装置において、命令ブレークが発生した場合、特定用途演算命令によるオーバーフローなどの演算例外が発生したことを検出し、例外処理をおこなうことができる。したがって、デバッグ作業の労力が軽減されるとともに、デバッグ作業に要する時間が短縮されるので、その結果、プログラムの開発期間が短縮されるという効果が得られる。

【 0 0 8 0 】

(実施の形態 4)

図 1 3 は、本発明の実施の形態 4 にかかる特定用途演算命令の例外処理方法を適用した情報処理装置の構成を示すブロック図である。この情報処理装置が実施の形態 1 と異なるのはつぎの 3 点である。すなわち、第 1 に、メモリ 2 1 0 に特定用途演算例外ブレークポイント・テーブル（図 1 3 では省略）が記憶されていることである。

【 0 0 8 1 】

第 2 に、実施の形態 1 の命令実行部 2 3 0 の代わりに設けられた命令実行部 5 3 0 には動作モード読み書き命令実行部 2 4 1 が設けられていないことである。第 3 に、実施の形態 1 のレジスタ制御部 2 6 0 の代わりに設けられたレジスタ制御部 4 6 0 に、動作モード・レジスタ 2 6 8 が設けられていないことである。

【 0 0 8 2 】

その他の構成については、実施の形態 1 と同様であるため、実施の形態 1 と同様の構成については、実施の形態 1 と同じ符号を付して説明を省略する。なお、実施の形態 4 では、上述した実施の形態 1 の説明中の動作モード読み書き命令実行部 2 4 1 および動作モード・レジスタ 2 6 8 に関する説明は適用されないものとする。

【 0 0 8 3 】

図 1 4 に、特定用途演算例外ブレークポイント・テーブルの一例を示す。特定用途演算例外ブレークポイント・テーブル 2 1 1 は、特定用途演算命令の実行中に発生する演算例外の検出をおこなうブレークポイントを示すテーブルである。

特定用途演算例外ブレークポイント・テーブル 2 1 1 には、命令ブレークおよびソフトウェア・ブレークによるブレークポイントのうち、特定用途演算命令の実行中に発生した演算例外を検出する対象となるブレークポイントが登録される。特定用途演算例外ブレークポイント・テーブル 2 1 1 の命令アドレス欄 2 1 2 は、ブレークポイントの命令アドレスを示す欄である。

【 0 0 8 4 】

前記命令実行部 5 3 0 は、命令デコード部 2 3 1、ロード命令実行部 2 3 2、ストア命令実行部 2 3 3、演算命令実行部 2 3 4、その他の命令の実行部 2 3 5、浮動小数点ロード命令実行部 2 3 6、浮動小数点ストア命令実行部 2 3 7、浮動小数点演算命令実行部 2 3 8、特定用途演算命令実行部 2 3 9、および特定用途演算ステータス読み書き命令実行部 2 4 0 を備えている。

【 0 0 8 5 】

前記レジスタ制御部 4 6 0 は、COND 2 6 1、復帰命令アドレス・レジスタ 2 6 2、動作レジスタ 2 6 3、割り込み発生前動作レジスタ 2 6 4、汎用レジスタ 2 6 5、浮動小数点レジスタ 2 6 6、および特定演算ステータス・レジスタ 2 6 7 を備えている。

【 0 0 8 6 】

つぎに、本発明の実施の形態 4 にかかる特定用途演算命令の例外処理方法について説明する。図 1 5 は、本発明の実施の形態 4 にかかる特定用途演算命令の例外処理方法のうち、ソフトウェア・ブレーク法による割り込み処理プログラムのフローチャートである。

【 0 0 8 7 】

図 1 5 のフローチャートにおいて、プログラムの実行が中断され、割り込み処理プログラムが開始されると、まずコンテキストを退避する（ステップ S 1 5 0 1）。そして、メモリ 2 1 0 に記憶された特定用途演算例外ブレークポイント・テーブル 2 1 1 にブレークポイントが含まれているか否かを確認する（ステップ S 1 5 0 2）。その結果、特定用途演算例外ブレークポイント・テーブル 2 1 1 に含まれていない場合（ステップ S 1 5 0 3：含まれていない）は、ソフトウェア・ブレーク処理をおこなった（ステップ S 1 5 0 4）後、コンテキストを復元

し（ステップ S 1 5 0 5）、割り込み復帰命令により割り込みから復帰し（ステップ S 1 5 0 6）、割り込み処理を終了する。

【 0 0 8 8 】

一方、特定用途演算例外ブレークポイント・テーブル 2 1 1 にブレークポイントが含まれている場合（ステップ S 1 5 0 3：含まれている）は、特定用途演算ステータス・レジスタ 2 6 7 に対する読み出し命令により特定用途演算ステータス・レジスタ 2 6 7 の値を読み出し、特定用途演算命令の実行中の演算例外を検出済みであるか否かを確認する（ステップ S 1 5 0 7）。その結果、演算例外を未検出であれば（ステップ S 1 5 0 8：演算例外未検出）、ステップ S 1 5 0 4 へ移行し、ステップ S 1 5 0 4 ～ S 1 5 0 6 の処理をおこない、割り込み処理を終了する。これに対し、演算例外を検出済みであれば（ステップ S 1 5 0 8：演算例外検出済み）、特定用途演算例外処理をおこなった（ステップ S 1 5 0 9）後、ステップ S 1 5 0 4 へ移行し、ステップ S 1 5 0 4 ～ S 1 5 0 6 の処理をおこない、割り込み処理を終了する。

【 0 0 8 9 】

図 1 6 は、本発明の実施の形態 4 にかかる特定用途演算命令の例外処理方法のうち、命令ブレーク法による割り込み処理プログラムのフローチャートである。プログラムの実行が中断され、割り込み処理プログラムが開始されると、まずコンテキストを退避する（ステップ S 1 6 0 1）。そして、メモリ 2 1 0 に記憶された特定用途演算例外ブレークポイント・テーブル 2 1 1 にブレークポイントが含まれているか否かを確認する（ステップ S 1 6 0 2）。その結果、特定用途演算例外ブレークポイント・テーブル 2 1 1 に含まれていない場合（ステップ S 1 6 0 3：含まれていない）は、命令ブレーク処理をおこなった（ステップ S 1 6 0 4）後、コンテキストを復元し（ステップ S 1 6 0 5）、割り込み復帰命令により割り込みから復帰し（ステップ S 1 6 0 6）、割り込み処理を終了する。

【 0 0 9 0 】

一方、特定用途演算例外ブレークポイント・テーブル 2 1 1 にブレークポイントが含まれている場合（ステップ S 1 6 0 3：含まれている）は、特定用途演算ステータス・レジスタ 2 6 7 に対する読み出し命令により特定用途演算ステータ

ス・レジスタ 2 6 7 の値を読み出し、特定用途演算命令の実行中の演算例外を検出済みであるか否かを確認する（ステップ S 1 6 0 7）。その結果、演算例外を検出済みであれば（ステップ S 1 6 0 8：演算例外検出済み）、ステップ S 1 6 0 4 へ移行し、ステップ S 1 6 0 4 ～ S 1 6 0 6 の処理をおこない、割り込み処理を終了する。これに対し、演算例外を検出済みでなければ（ステップ S 1 6 0 8：演算例外未検出）、ステップ S 1 6 0 4 へ移行し、ステップ S 1 6 0 4 ～ S 1 6 0 6 の処理をおこない、割り込み処理を終了する。

【 0 0 9 1 】

実施の形態 4 におけるステップ・ブレイク法による割り込み処理プログラムは、図 1 9 に示す従来のフローチャートと同様であるため、図示および説明を省略する。

【 0 0 9 2 】

上述した実施の形態 4 によれば、特定用途演算命令を有する情報処理装置において、ソフトウェア・ブレイクまたは命令ブレイクが発生した場合、特定用途演算命令によるオーバーフローなどの演算例外が発生したことを検出し、例外処理をおこなうことができる。したがって、デバッグ作業の労力が軽減されるとともに、デバッグ作業に要する時間が短縮されるので、プログラムの開発期間が短縮されるという効果が得られる。

【 0 0 9 3 】

【発明の効果】

本発明によれば、特定用途演算命令を有する情報処理装置において、ステップ・ブレイク、ソフトウェア・ブレイクまたは命令ブレイクの発生時に、特定用途演算命令によるオーバーフローなどの演算例外が発生したことを検出し、例外処理をおこなうことができる。したがって、デバッグ作業の労力が軽減されるとともに、デバッグ作業に要する時間が短縮されるので、プログラムの開発期間が短縮されるという効果が得られる。

【図面の簡単な説明】

【図 1】

本発明の実施の形態 1 にかかる特定用途演算命令の例外処理方法を適用した情報処理装置の構成を示すブロック図である。

【図 2】

本発明の実施の形態 1 における命令ブレイク検出部の一例を示すブロック図である。

【図 3】

本発明の実施の形態 1 における動作レジスタの一例を示す構成図である。

【図 4】

本発明の実施の形態 1 における割り込み発生前動作レジスタの一例を示す構成図である。

【図 5】

本発明の実施の形態 1 にかかる特定用途演算命令の例外処理方法のうち、ステップ・ブレイク法による割り込み処理プログラムのフローチャートである。

【図 6】

本発明の実施の形態 1 にかかる特定用途演算命令の例外処理方法のうち、ソフトウェア・ブレイク法による割り込み処理プログラムのフローチャートである。

【図 7】

本発明の実施の形態 1 にかかる特定用途演算命令の例外処理方法のうち、命令ブレイク法による割り込み処理プログラムのフローチャートである。

【図 8】

本発明の実施の形態 2 にかかる特定用途演算命令の例外処理方法を適用した情報処理装置の構成を示すブロック図である。

【図 9】

本発明の実施の形態 2 にかかる特定用途演算命令の例外処理方法のうち、ソフトウェア・ブレイク法による割り込み処理プログラムのフローチャートである。

【図 1 0】

本発明の実施の形態 3 にかかる特定用途演算命令の例外処理方法を適用した情報処理装置の構成を示すブロック図である。

【図 1 1】

本発明の実施の形態 3 における命令ブレイク検出部の一例を示すブロック図である。

【図 1 2】

本発明の実施の形態 3 にかかる特定用途演算命令の例外処理方法のうち、命令ブレイク法による割り込み処理プログラムのフローチャートである。

【図 1 3】

本発明の実施の形態 4 にかかる特定用途演算命令の例外処理方法を適用した情報処理装置の構成を示すブロック図である。

【図 1 4】

本発明の実施の形態 4 における特定用途演算例外ブレイクポイント・テーブルの一例を示す構成図である。

【図 1 5】

本発明の実施の形態 4 にかかる特定用途演算命令の例外処理方法のうち、ソフトウェア・ブレイク法による割り込み処理プログラムのフローチャートである。

【図 1 6】

本発明の実施の形態 4 にかかる特定用途演算命令の例外処理方法のうち、命令ブレイク法による割り込み処理プログラムのフローチャートである。

【図 1 7】

従来の命令ブレイク法による割り込み処理プログラムのフローチャートである。

【図 1 8】

従来のソフトウェア・ブレイク法による割り込み処理プログラムのフローチャートである。

【図 1 9】

従来のステップ・ブレイク法による割り込み処理プログラムのフローチャートである。

【符号の説明】

2 1 0 メモリ

2 1 1 演 3 算例外検出対象のブレイクポイントを登録したテーブル（特定用

途演算例外ブレークポイント・テーブル)

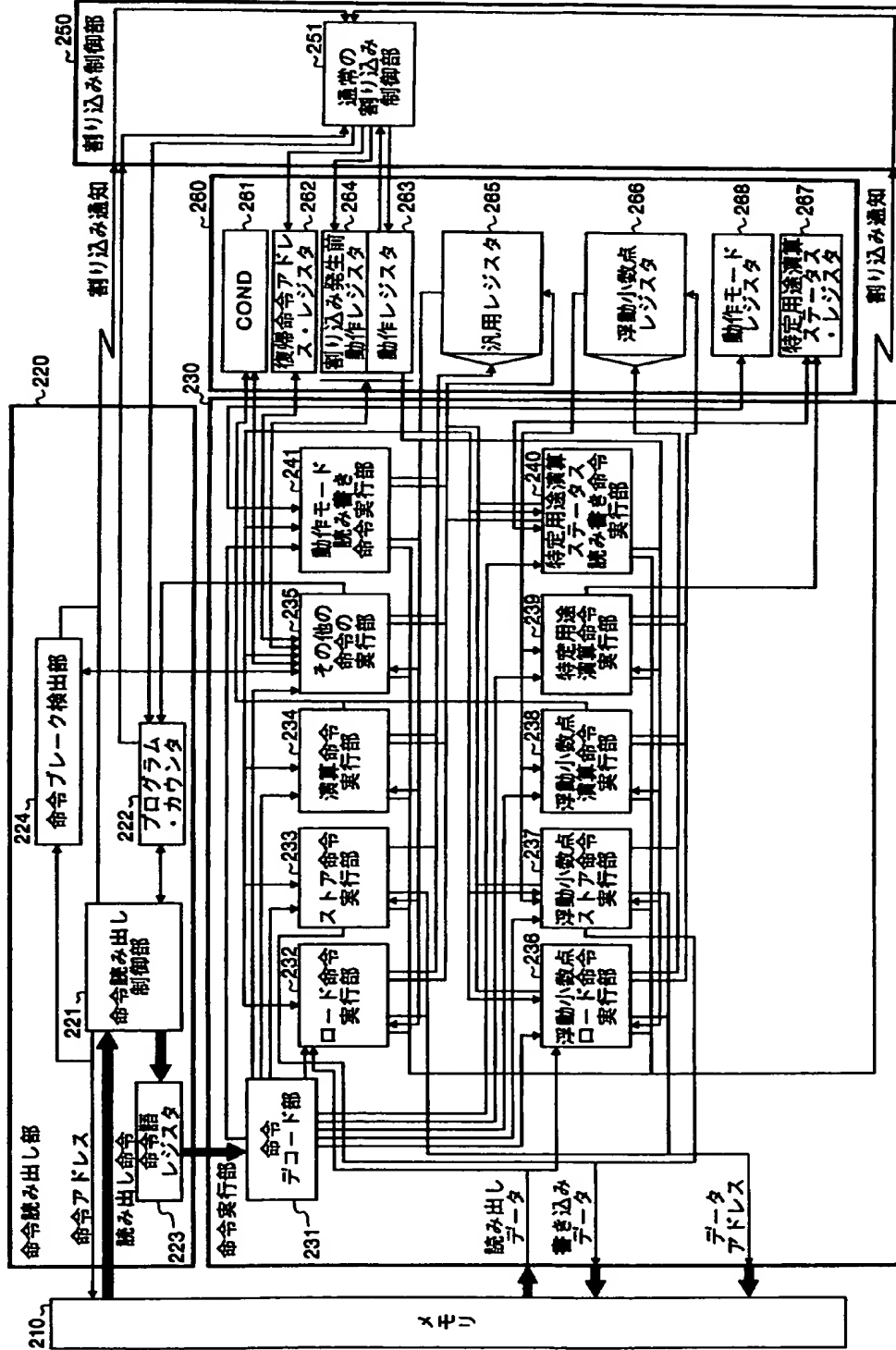
- 2 4 0 特定用途演算ステータス読み書き命令実行部
- 2 4 1 動作モード読み書き命令実行部
- 3 3 1 特定用途演算例外検出ブレークポイント命令の実行部
- 2 6 7 演算例外検出済みであることを記憶したレジスタ (特定用途演算ステータス・レジスタ)
- 2 6 8 演算例外検出状態であることを記憶したレジスタ (動作モード・レジスタ)
- 4 2 5 a ~ 4 2 5 d 演算例外検出対象の命令ブレーク発生アドレスを記憶したレジスタ (命令ブレークポイント・レジスタ)

【書類名】

図面

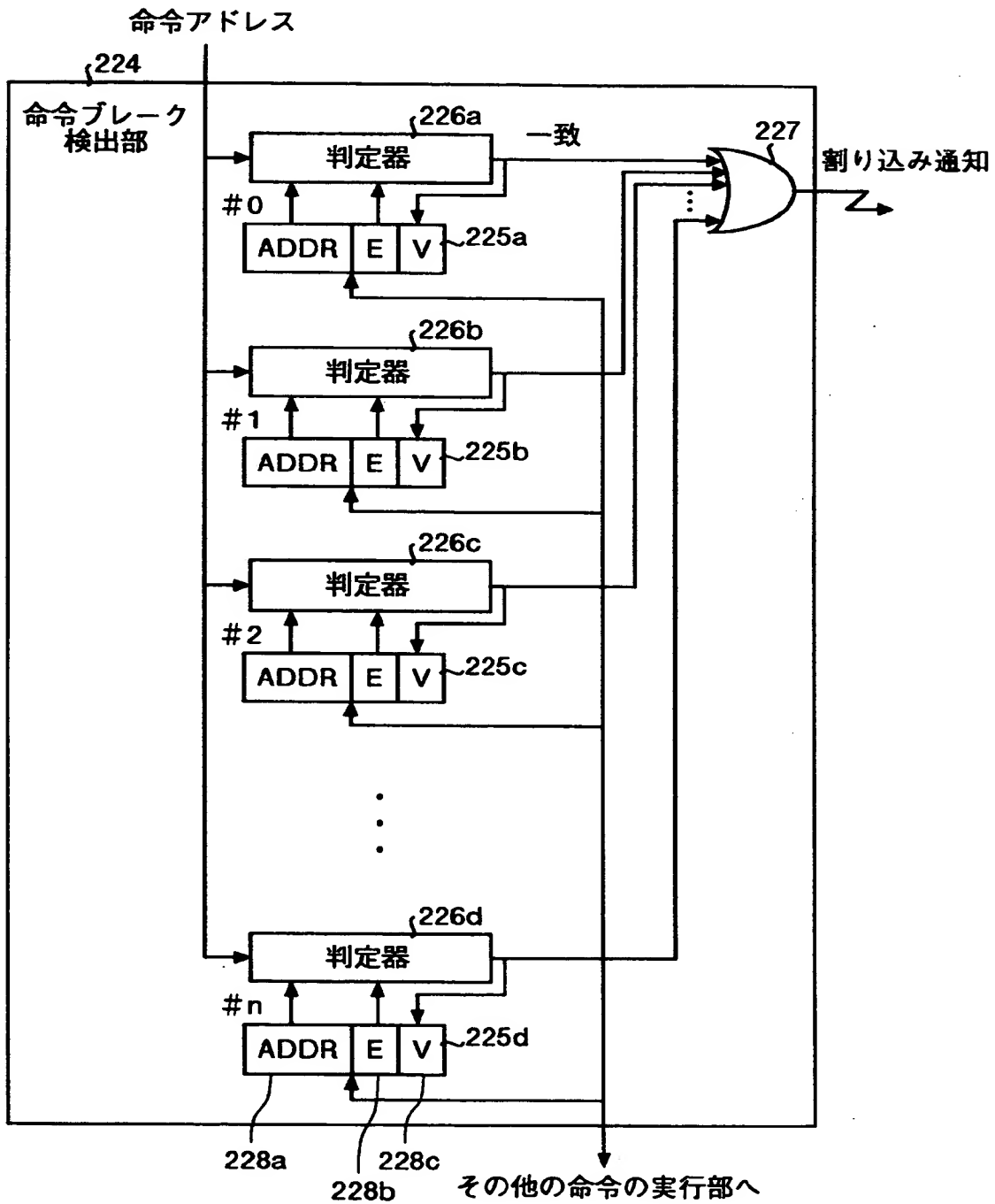
【図 1】

実施の形態 1 にかかる特定用途演算命令の例外処理方法を適用した情報処理装置の構成を示すブロック図



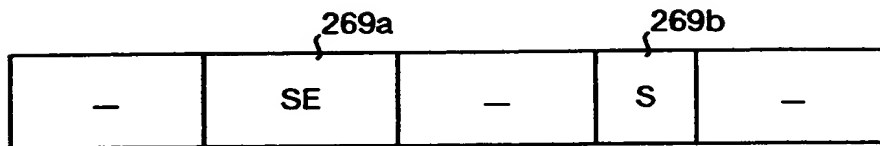
【図 2】

実施の形態 1 における命令ブレーク検出部の一例を示すブロック図



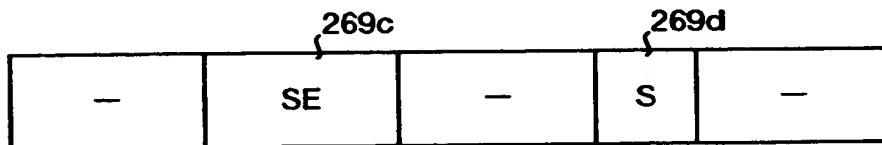
【図 3】

実施の形態 1 における動作レジスタの一例を示す構成図



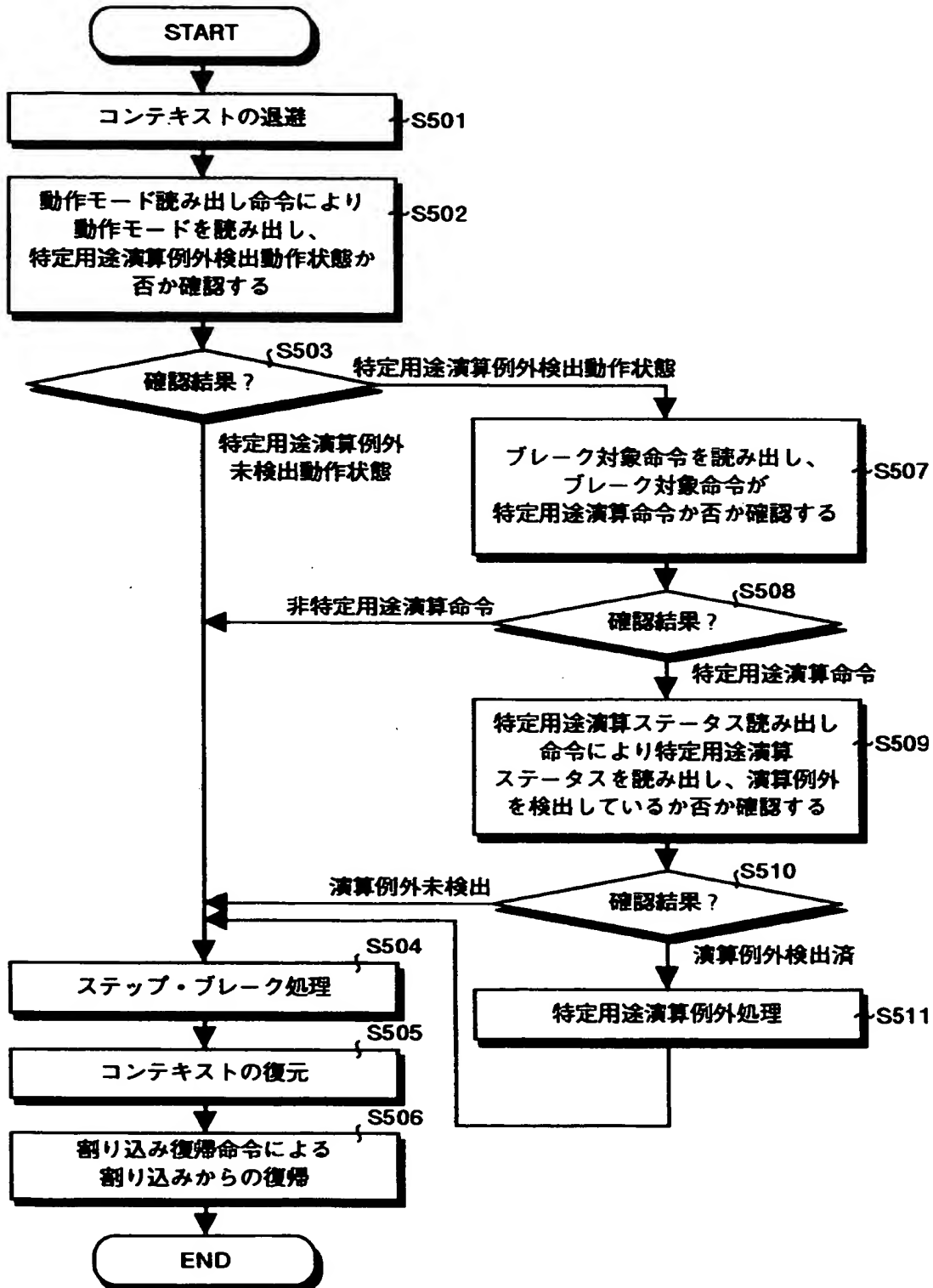
【図 4】

実施の形態 1 における割り込み発生前動作レジスタの一例を示す構成図



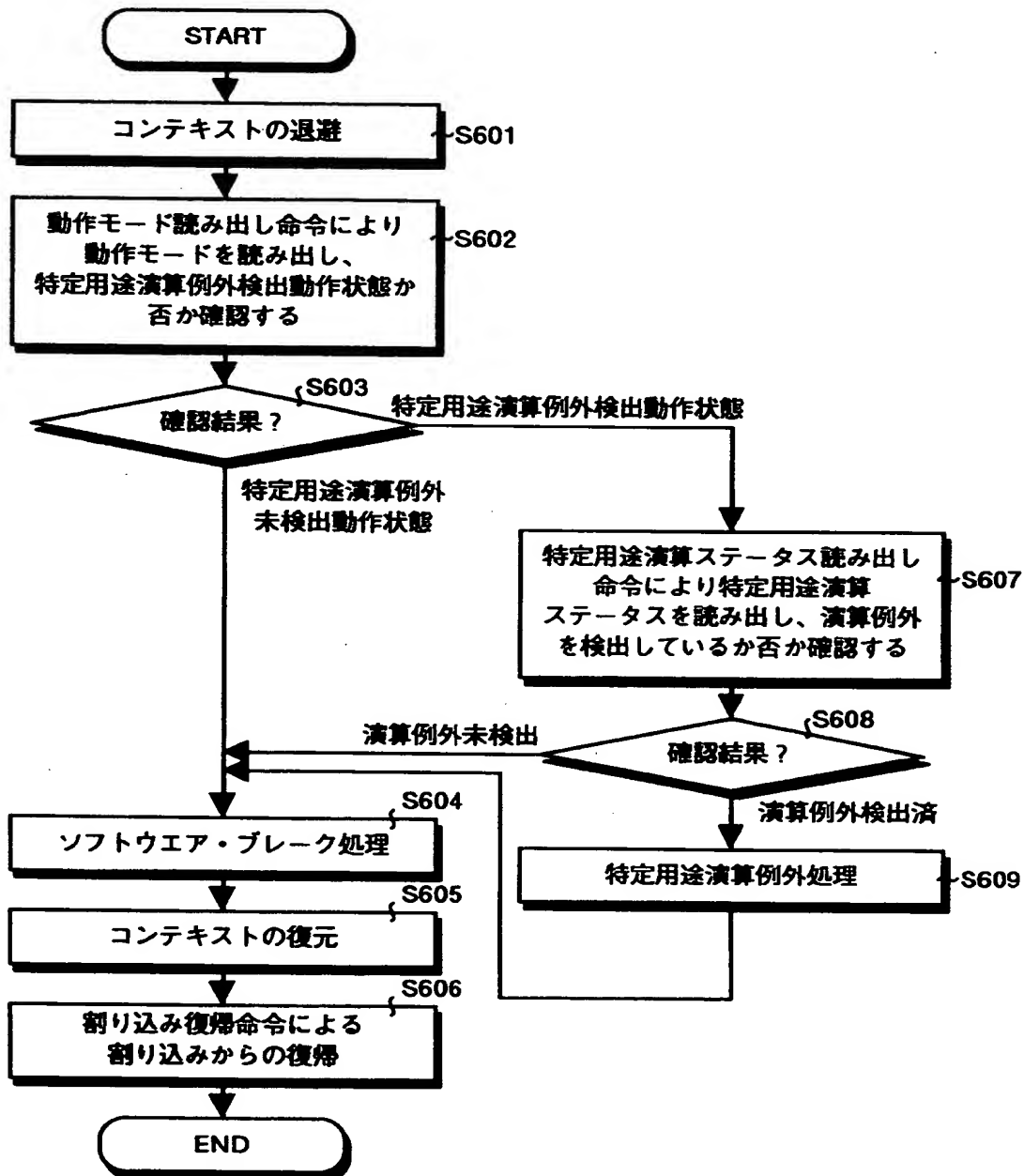
【図 5】

実施の形態 1 にかかる特定用途演算命令の例外処理方法のうち、
ステップ・ブレイク法による割り込み処理プログラムのフローチャート



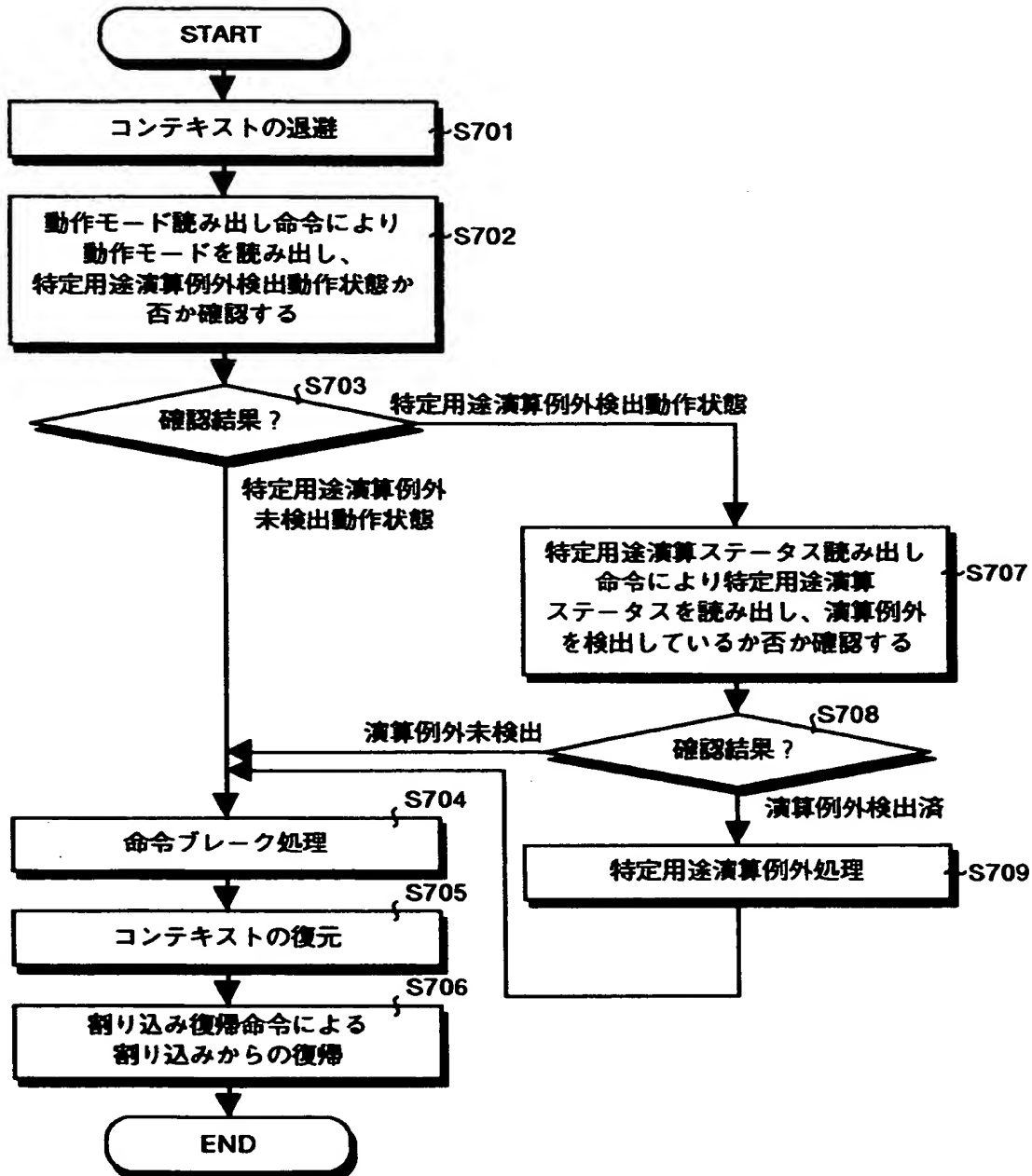
【図 6】

実施の形態 1 にかかる特定用途演算命令の例外処理方法のうち、
ソフトウェア・ブレイク法による割り込み処理プログラムのフローチャート



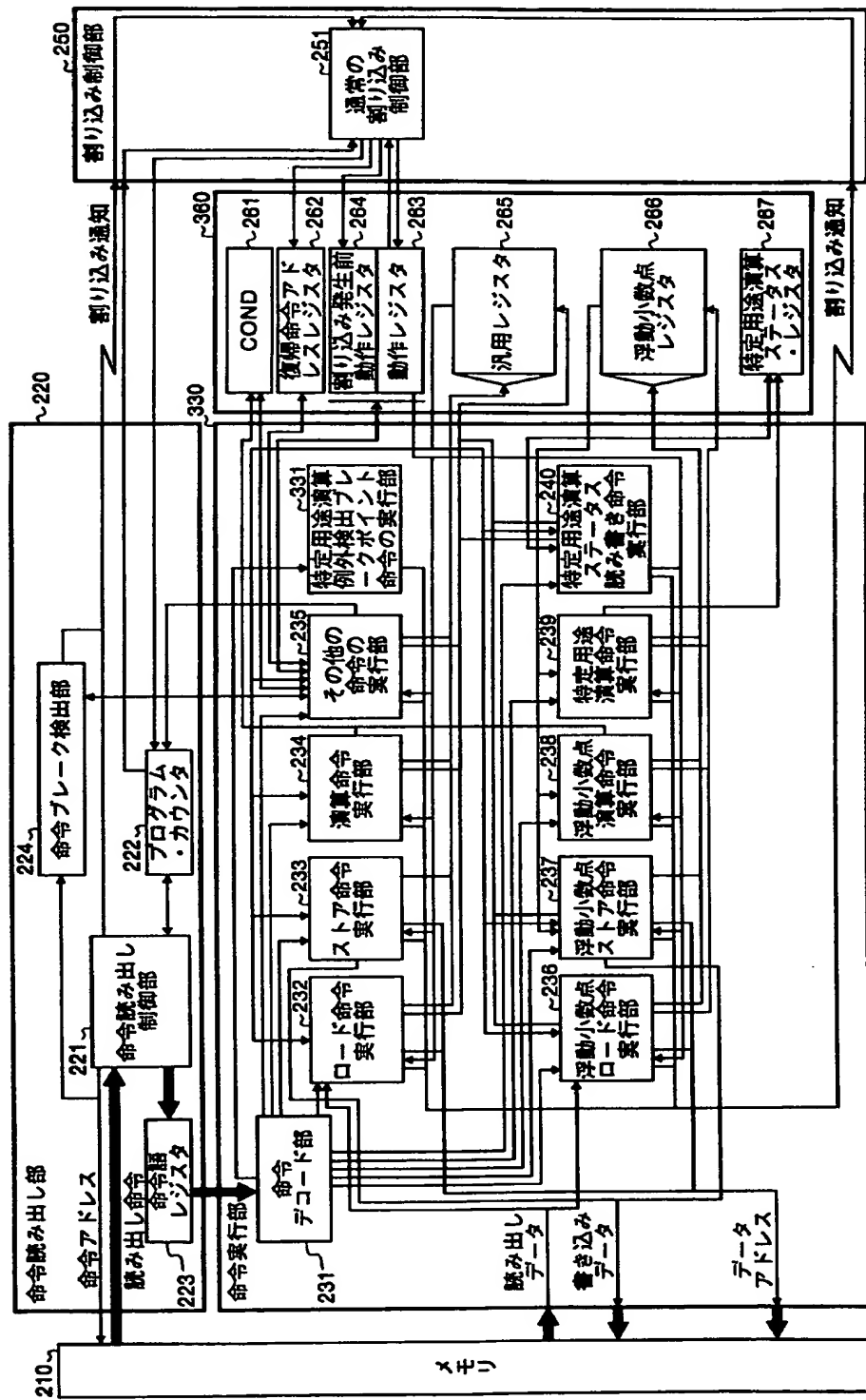
【図 7】

実施の形態 1 にかかる特定用途演算命令の例外処理方法のうち、
命令ブレイク法による割り込み処理プログラムのフローチャート



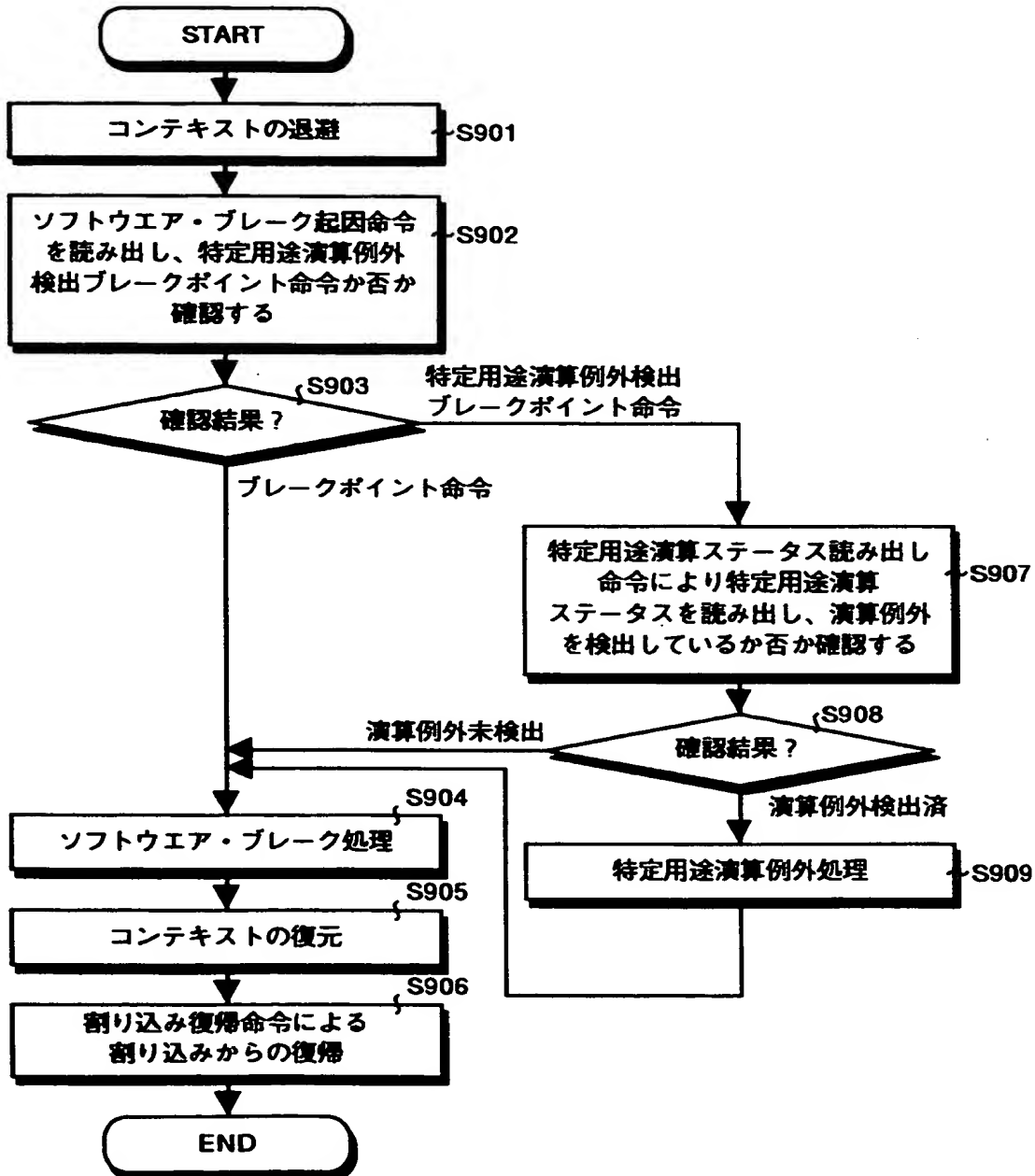
【図 8】

実施の形態 2 にかかる特定用途演算命令の例外処理装置の構成を示すブロック図



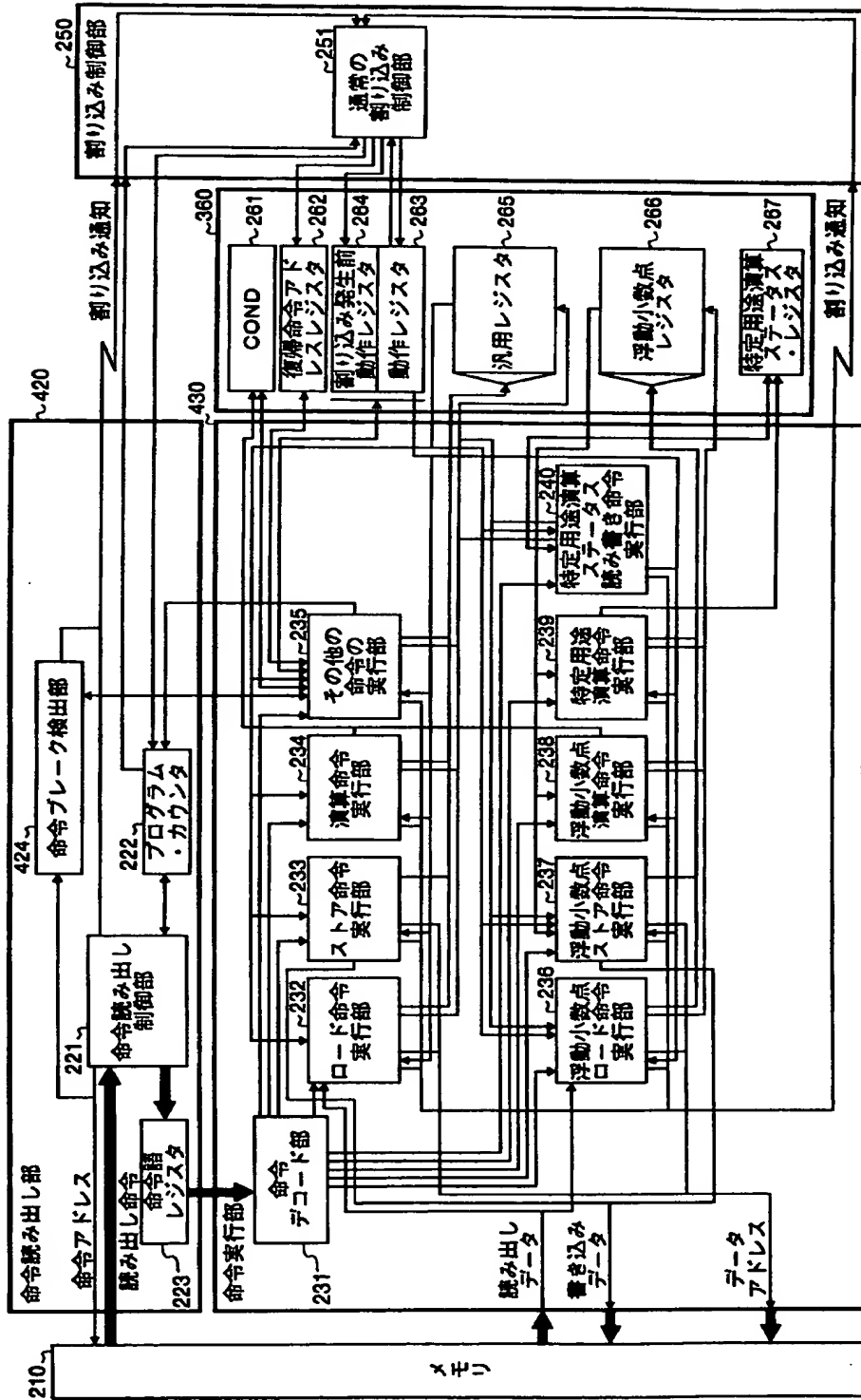
【図 9】

実施の形態 2 にかかる特定用途演算命令の例外処理方法のうち、
ソフトウェア・ブレイク法による割り込み処理プログラムのフローチャート



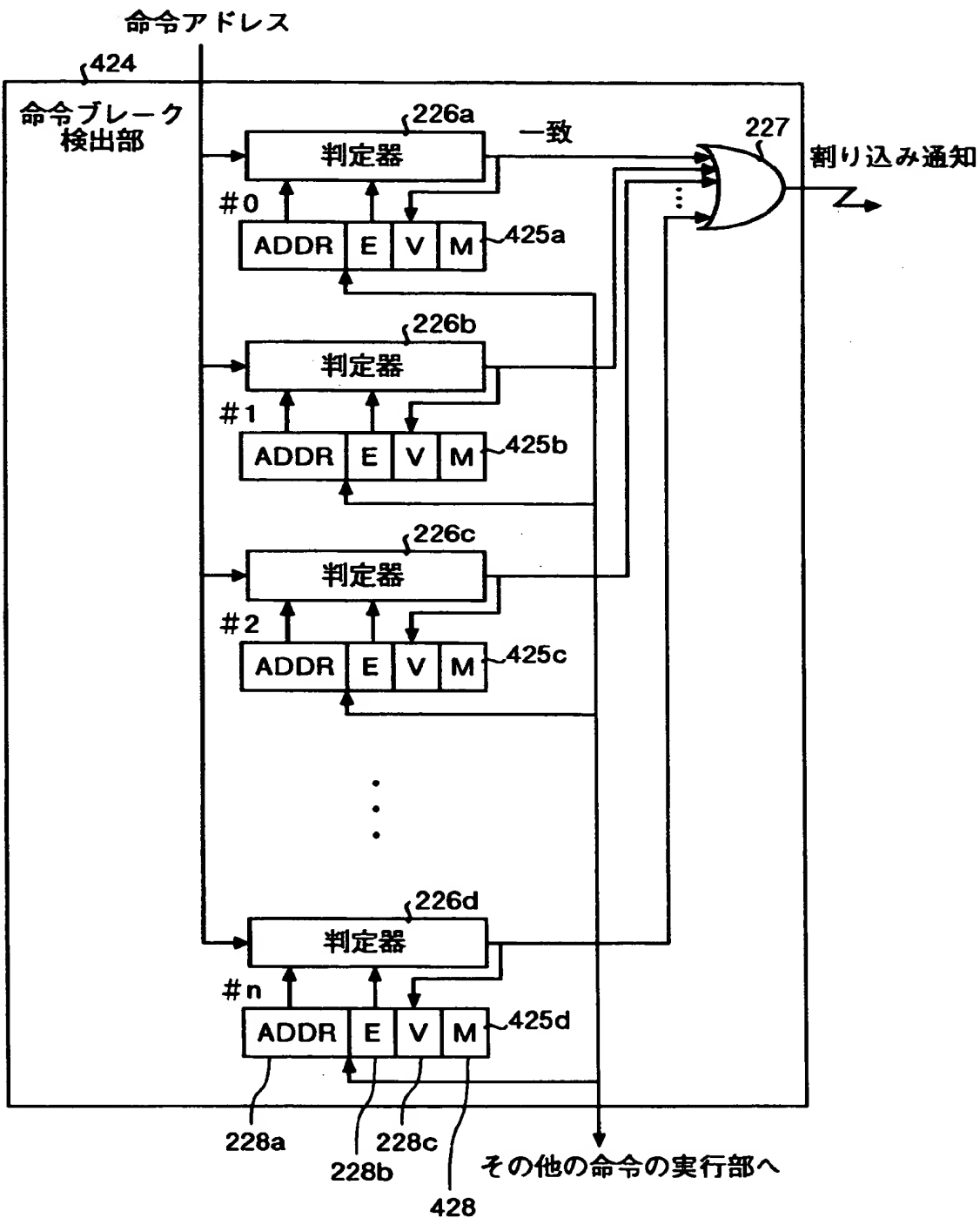
【図 10】

実施の形態 3 にかかる特定用途演算命令の例外処理方法を用いた情報処理装置の構成を示すブロック図



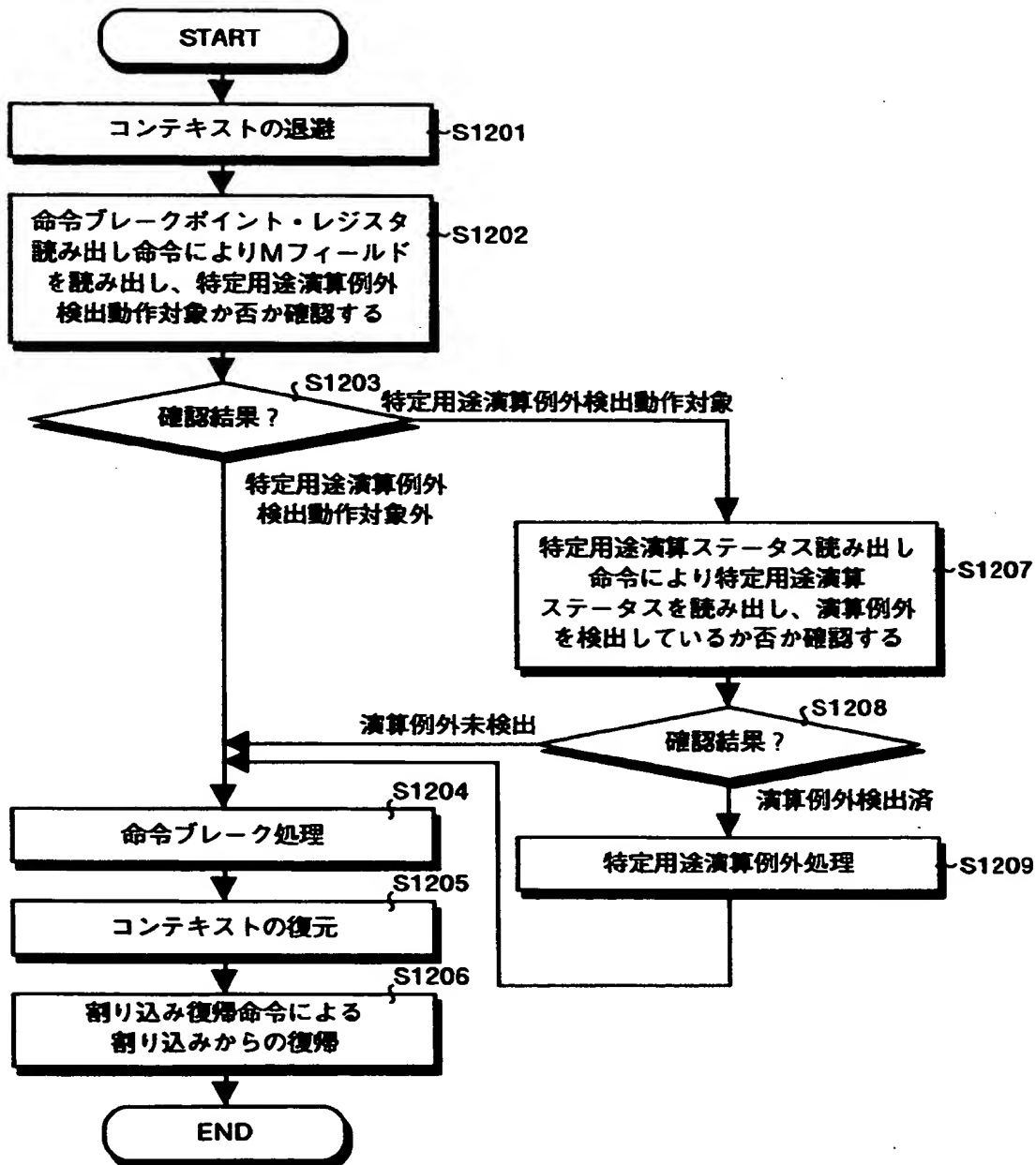
【図 1 1】

実施の形態 3 における命令ブレーク検出部の一例を示すブロック図



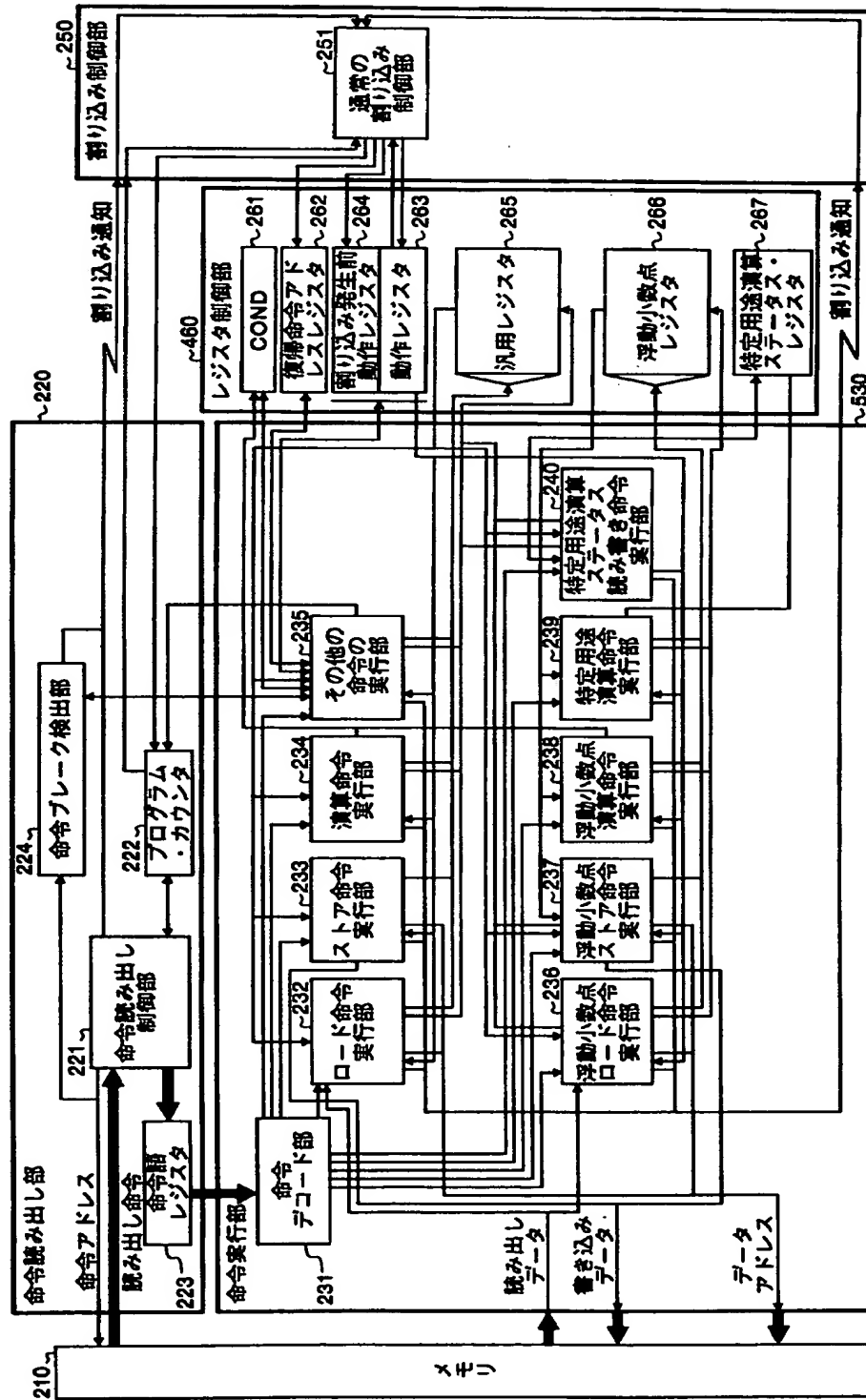
【図 1 2】

実施の形態 3 にかかる特定用途演算命令の例外処理方法のうち、
命令ブレイク法による割り込み処理プログラムのフローチャート



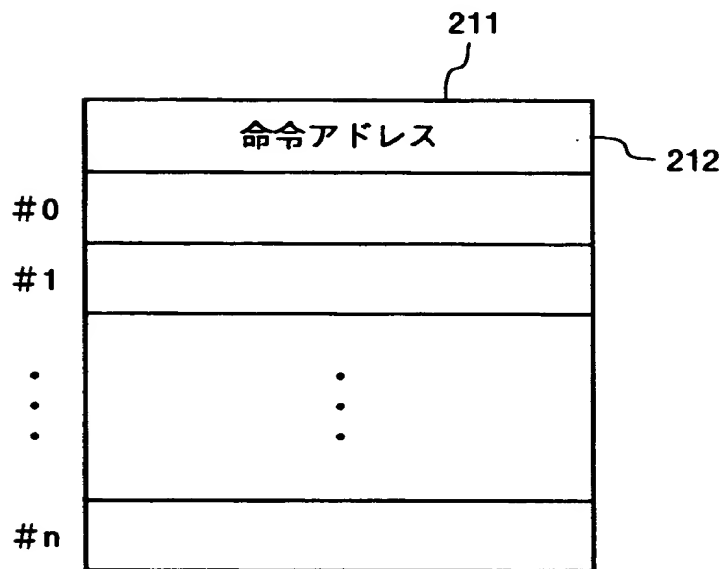
【図 13】

実施の形態 4 にかかる特定用途演算命令の例外処理方法を適用した情報処理装置の構成を示すブロック図



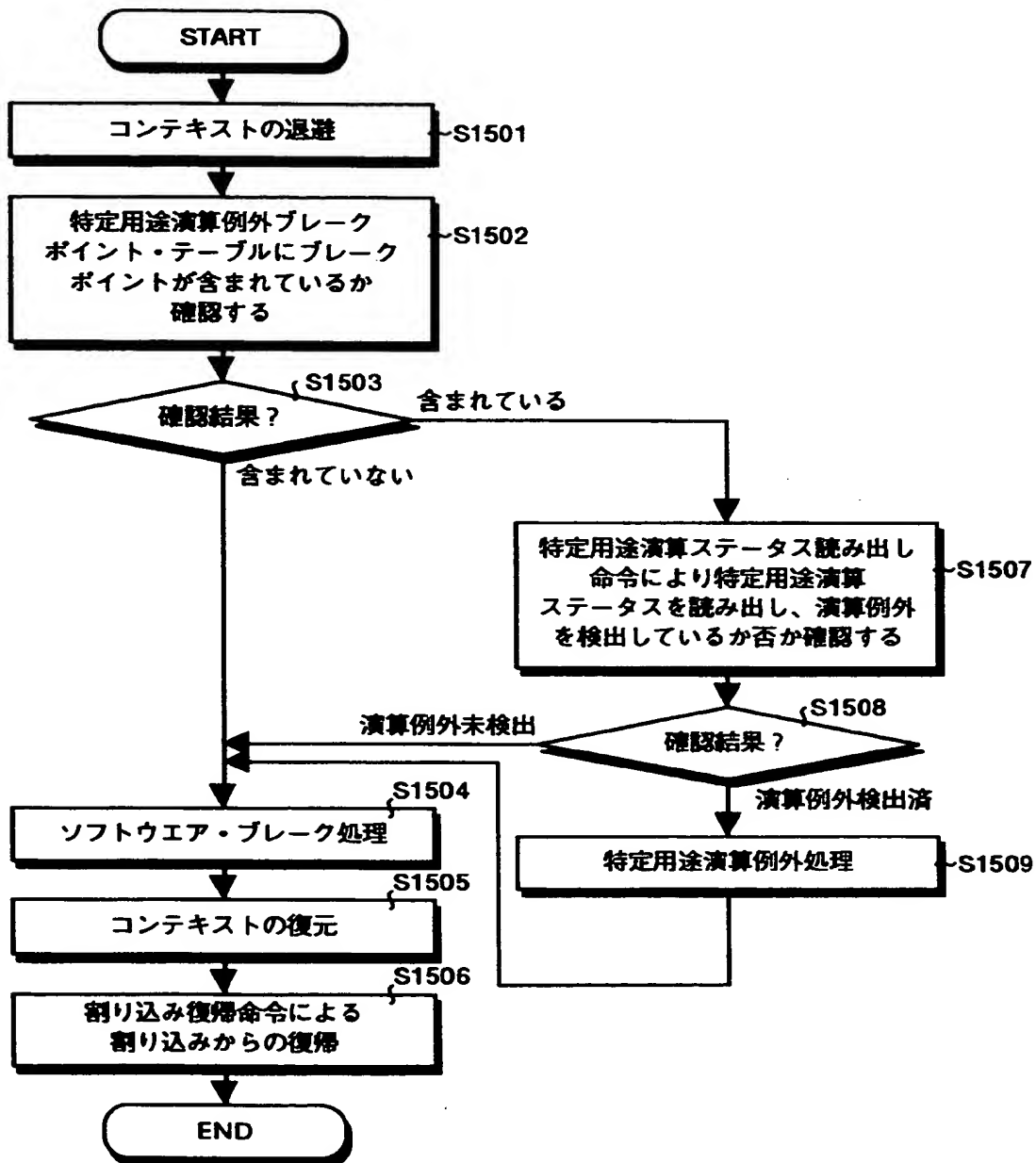
【図 1 4】

実施の形態 4 における特定用途演算例外
ブレークポイント・テーブルの一例を示す構成図



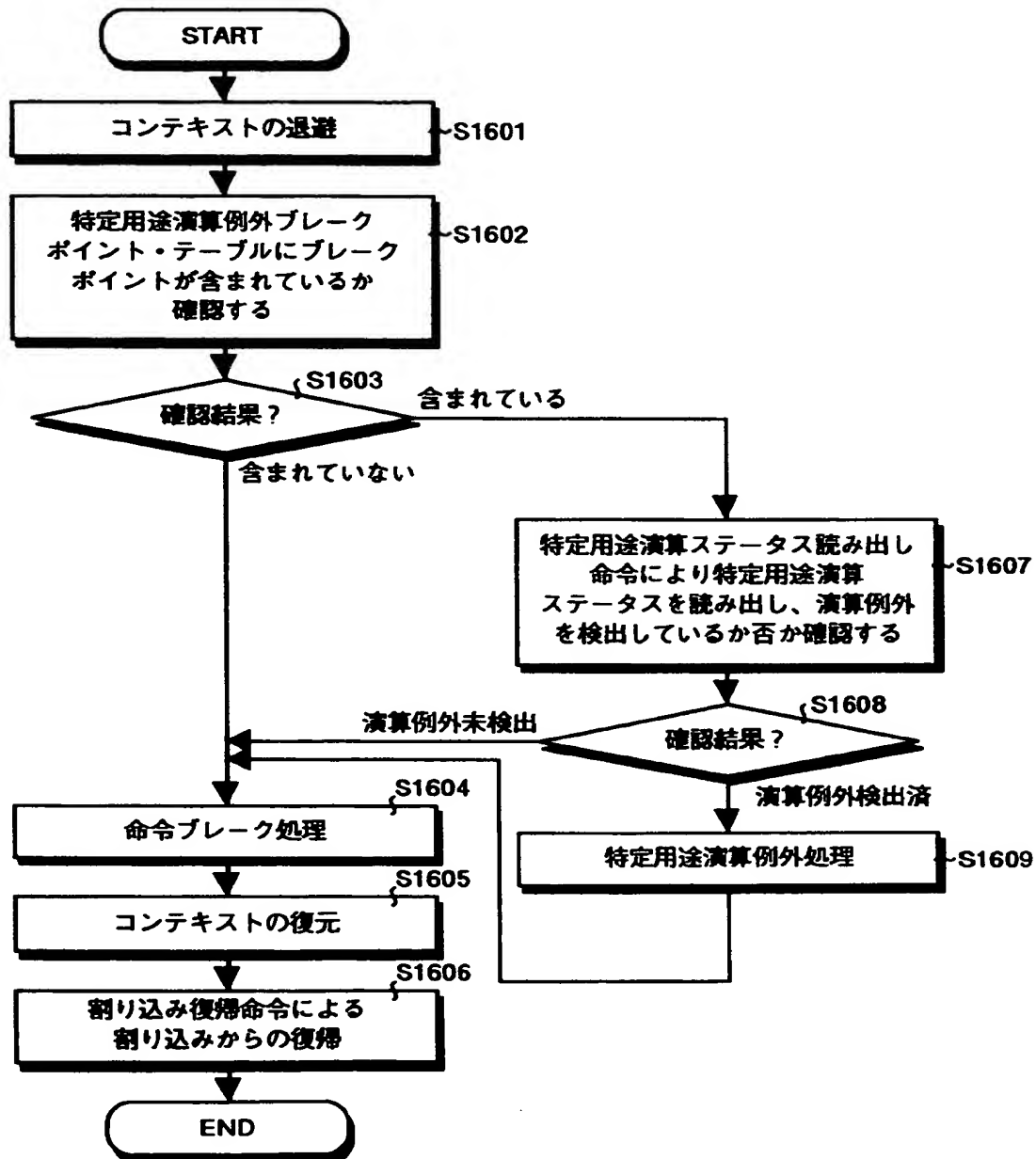
【図 15】

実施の形態 4 にかかる特定用途演算命令の例外処理方法のうち、
ソフトウェア・ブレイク法による割り込み処理プログラムのフローチャート



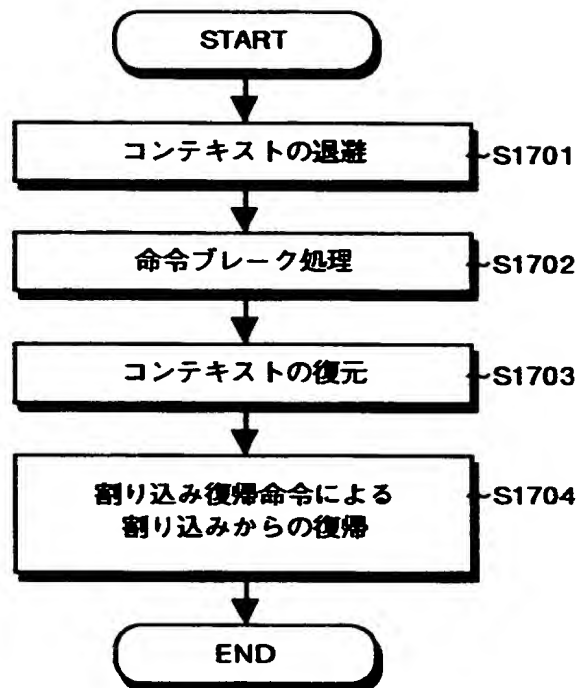
【図 1 6】

実施の形態 4 にかかる特定用途演算命令の例外処理方法のうち、
命令ブレイク法による割り込み処理プログラムのフローチャート



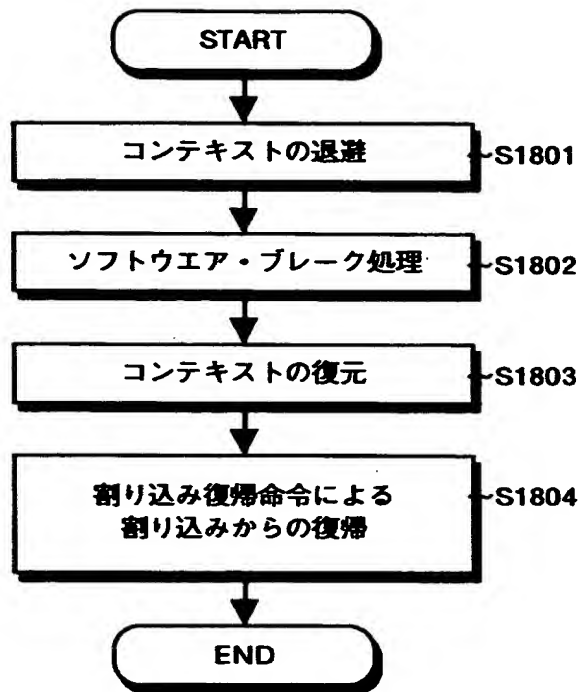
【図 1 7】

従来の命令ブレイク法による
割り込み処理プログラムのフローチャート



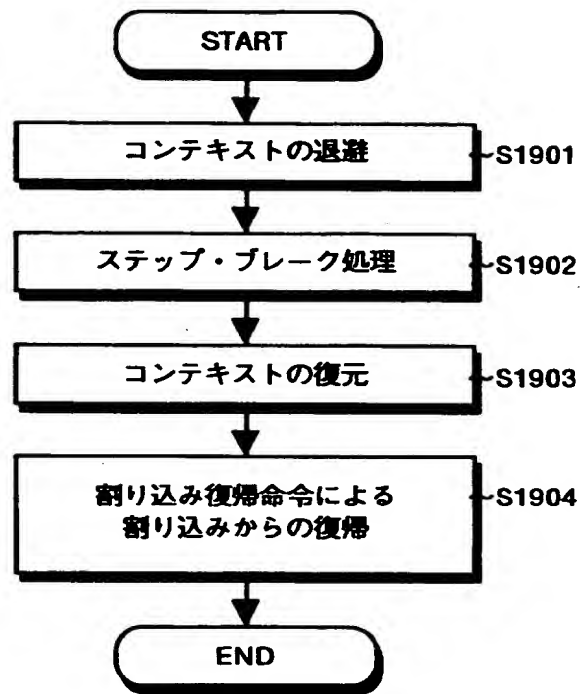
【図 1 8】

従来のソフトウェア・ブレイク法による
割り込み処理プログラムのフローチャート



【図 1 9】

従来のステップ・ブレイク法による
割り込み処理プログラムのフローチャート



【書類名】 要約書

【要約】

【課題】 特定用途演算命令を有する情報処理装置において、特定用途演算命令の実行中に発生した演算例外を検出し、例外処理をおこなうこと。

【解決手段】 プログラムの実行中断後、割り込み処理プログラムが開始されると、コンテキストを退避した後、特定用途演算命令の実行中に発生する演算例外の検出をおこなう状態に設定されているか否かを確認する。演算例外の検出をおこなう状態に設定されている場合には、さらに特定用途演算命令の実行中の演算例外を検出済みであるか否かを確認する。演算例外を検出済みであれば特定用途演算例外処理をおこなった後、ステップ・ブレイク処理、ソフトウェア・ブレイク処理または命令ブレイク処理をおこなう。そして、コンテキストを復元し、割り込みから復帰して終了する。

【選択図】 図 5

出 願 人 履 歴 情 報

識別番号 [0 0 0 0 0 5 2 2 3]

1. 変更年月日 1 9 9 6 年 3 月 2 6 日

[変更理由] 住所変更

住 所 神奈川県川崎市中原区上小田中4丁目1番1号

氏 名 富士通株式会社